

# Commodore

# WORLD

Nº 11 - ENERO 1985

315 PTAS.

... y todas  
nuestras secciones  
**MAGIA - COLABORACIONES**

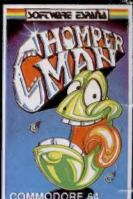
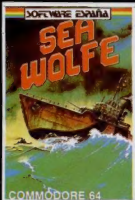
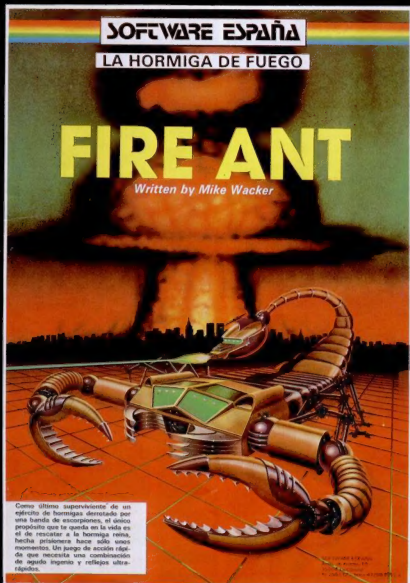
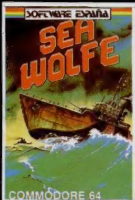
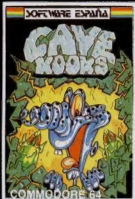
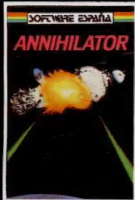
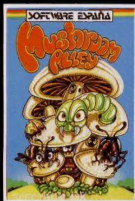
## Código máquina "a gogó"

**Descubrir el secreto de la estructura interior de los Commodore**  
**El arte de realizar organigramas**  
**Castellanizar el Commodore-64**  
**Rutinas Basic-3 (Paginación)**  
**Constructor de datos**  
**Utilidades y trucos**  
**El duelo**



# SOFTWARE ESPAÑA

SOFTWARE ESPAÑA  
Avenida de Arteijo, 19  
15004 La Coruña  
Teléf. (981) 25 51 72  
Télex 47206 PPLL-E



# Commodore WORLD

Commodore World  
está publicado por SIMSA  
y la colaboración  
de todos nuestros lectores

## EQUIPO:

Manuel AMADO;  
Juan L. AMESTI; Nieves CHESA;  
José Luis ERRAQUIN;  
M.<sup>a</sup> Jesús GARCÍA;  
Miguel A. HERMOSELL;  
Alvaro IBÁÑEZ; María LOPEZ;  
Juan MARQUEZ; Juan MARTINEZ;  
Pere MASATS; Victoria MORALES;  
Rafael PARDO; Diego ROMERO;  
Albert SANGLAS; Jordi SASTRE;  
Valerie SHANKS; Francisco ZABALA...  
...Y NUESTROS LECTORES

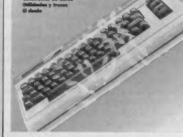
## SIMSA

Coordinador María López  
c/Barquillo, 21-3º Izda.  
28004 Madrid. Teléf.: (91) 231 23 88/95  
DELEGACIÓN EN BARCELONA:  
Sant Gervasi de Cassoles, 39 despacho 4.  
08022 Barcelona  
Tels.: (93) 212 73 45/212 88 48  
Colabora MEC-COMMODORE  
con Club Commodore  
Coordinador Pere Masats  
Valencia, 49-53 - 08015 Barcelona  
Teléf.: (93) 325 50 08

## Commodore WORLD

### Código máquina "a gogó"

Reservados todos los derechos de la explotación editorial de los Commodore  
El uso de cualquier información  
Contribuyente al Commodore-64  
Incluye: Búsqueda y Programación  
Contribuyente de Datos  
Contribuyente y Programación  
El uso de



PROHIBIDA LA  
REPRODUCCIÓN TOTAL O  
PARCIAL DE LOS  
ORIGINALES DE ESTA  
REVISTA SIN AUTORIZACIÓN  
HECHA POR ESCRITO.

NO NOS HACEMOS  
RESPONSABLES DE LAS  
OPINIONES EMITIDAS POR  
NUESTROS COLABORADORES.

## Imprime:

IBERDOS, S. A.  
Germán Pérez Carrasco, 24.  
28027 Madrid

Déposito Legal: M-2944-1984

# SUMARIO

CONTENIDO	PAG.
SUPERINTERESANTISIMO	4
CONSTRUCTOR DE DATOS	8
DESCUBRIR EL SECRETO DE LA ESTRUCTURA INTERIOR DE LOS COMMODORES	12
CASTELLANIZAR EL COMMODORE-64	18
EL ARTE DE REALIZAR ORGANIGRAMAS	22
RUTINAS BASIC-III-PAGINACION	32
UNA SERIE DE UTILIDADES Y TRUCOS	36
COMPARTIENDO EXPERIENCIAS ENTRE AMIGOS	
Golf .....	38
El Concierto de Aranjuez .....	40
Batalla .....	45
MAGIA .....	52
CARTA BLANCA Y SEAMOS PREGUNTONES .....	55
DIRECTORIO	49
COMENTARIOS COMMODORE	51
VIDEOCASINO	
El duelo .....	58
CURSOLO DE LENGUAJE MAQUINA	61
MARKETCLUB	73



## PROXIMO NUMERO

### ESPECIAL IMPRESORAS

Cómo utilizar Centronics para los C-64 y los VIC.

Dibujo de una senoide con una impresora de margarita.

Caracteres españoles en impresoras OKI.

Dos grandes juegos: Cluedo y Cuevas de Alfa-ceti.

Programa de estadística... con deporte!!

Sumando con los "peques".

y todas vuestras colaboraciones.



Commodore World es miembro de CW Communications/Inc., el grupo editorial más grande y de mayor prestigio en el mundo en lo que se refiere al ámbito informático. Dicho grupo tiene a su cargo la edición de 52 publicaciones relacionadas con los ordenadores en 19 países. El grupo editorial está integrado por:

ALEMANIA: Computerwoche, Micro Computerwelt, PC Welt, Software Markt, CW Edition/Seminar, Computer Business, Run (Commodore);  
ARABIA SAUDI: Saudi Computerworld. ARGENTINA: Computerworld/Argentina. AUSTRALIA: Australasian Computerworld, Micro Computer Magazine, PC World. BRASIL: DataNews, MicroMundo. DINAMARCA: Computerworld/Danmark, Micro Verden. ESPAÑA: Computerworld/España, MicroSistema, Commodore World. ESTADOS UNIDOS: Computerworld, Hot CoCo, Incider, InfoWorld, Microcomputing, PC World, 80-Micro, Mac World Micro Market Word, PC Jr. World Run (Commodore). FINLANDIA: Mikro. FRANCIA: Le Monde Informatique. HOLANDA: Computerworld Benelux, Micro/Info. ITALIA: Computerworld Italia. JAPON: Computerworld Japan, Perso Comworld, PC World. MEXICO: Computerworld/México, Compumundo. NORUEGA: Computerworld Norge, Mikro Data. REINO UNIDO DE GRAN BRETAÑA: Computer Management, Computer Business Europe. REPUBLICA POPULAR DE CHINA: China Computerworld. SINGAPUR: Asian Computerworld. SUECIA: ComputerSweden, MikroDatorn, Min Hemdator.

# SUPER-INTERESANTISIMO

NUMERO EXTRAORDINARIO DE PROGRAMAS 2.<sup>a</sup> EDICION

Nuestro número Extra de Programas ha tenido un éxito monstruoso y sin apenas haber llegado a la calle se ha agotado. Se está preparando una segunda edición ya que las peticiones del mismo nos siguen llegando masivamente y los kioscos todavía no se han casi estrenado. Incluimos al pie de esta página un boletín de reserva para esta segunda edición, ¡¡CUIDADO!! los que ya hayáis enviado el boletín anterior, no es necesario que repitáis el envío, dentro de unos días tendréis el ejemplar en vuestra casa.

#### FORMA DE PAGO DEL EXTRAORDINARIO CON BOLETIN DE RESERVA

Solamente se aceptan peticiones que vengan acompañadas de cheque o giro. No se aceptan reembolsos.

#### BIENVENIDA A LA RADIOAFICION AL MUNDO COMMODORIANO Y DESPEDIDA AL MISTERIO DE LOS POKES

La serie el misterio de los pokes está tocando a su fin como ya habréis podido ver. Para ocupar el espacio que dejará, ER-4-ALI (que no es otro que Diego), ha preparado una serie de tres capítulos destinada a las aplicaciones del VIC-20 y el C-64 en el campo de la radioafición. Los que sean radioaficionados ya sabrán que su equipo se puede utilizar para recibir men-

sajes, y por supuesto, para enviarlos al éter. Se puede trabajar con el ordenador personal en las modalidades de CW y RTTY (telegrafía y radio-teletipo), ampliando con ello el campo de aplicaciones del ordenador. También se puede utilizar para calcular el rumbo ortodómico o trayectoria más corta entre dos puntos, orientación y diseño de antenas, etc. Los que no sean radioaficionados, quizás les despierta el gusanillo y comienzan a interesarse un poco por ese mundo tan curioso de las comunicaciones por radio, vía satélite, etc.

El motivo de escoger este tema ha sido la cantidad de cartas recibidas en la redacción solicitándolo. Para dar una idea de la difusión de esta afición en nuestro país, diremos que el número de aficionados con licencia oficial del ministerio pasa en estos momentos de treinta mil, y aproximadamente un cincuenta por ciento de ellos tienen también ordenadores personales.

Esperamos que la serie sea de interés para muchos de nuestros lectores.

#### RAMI Y ROMO AGREDIDOS POR EL CURSILLO DE CODIGO MAQUINA

Como veréis, en este número, por razones de espacio nos vemos obligados a quitar el comic que habitualmente aparece en nuestras páginas dirigido a los niños de dos a noventa años. La culpa de este desaguisado la ha tenido Diego, que nos ha inva-

dido doce páginas de la revista con su cursillo de lenguaje máquina. Esperamos que los asiduos de esta sección no le esperen en la calle un día de estos.

En el próximo número volverán nuestros pequeños amigos con sus aventuras, y ya hemos dado una buena reprimenda al autor de esta "fechoría" para que no vuelva a "pasarse de páginas".

#### IMPRESORAS PARA TODOS LOS COMMODORES

El próximo número va dedicado particularmente a las aplicaciones de impresoras.

En realidad lo que pasa es que aparecerán dos artículos dedicados a la utilización de impresoras que están dotadas de interfaz paralelo Centronics con los equipos Commodore (el VIC-20 y el C-64).

Como ya sabréis muchos, el tipo de interfaz Centronics es el que utilizan la mayoría de las impresoras del mercado, siendo por lo general el más barato dentro de las posibles opciones de conexión de cada modelo de impresora (RS-232, Centronics, etc.) por lo que se ha convertido en uno de los más utilizados por impresoras (si no es el más utilizado).

Con los dos programas que aparecerán podréis conectar otras impresoras a vuestros equipos sin la limitación de la gama de que dispone la casa Commodore. Además Diego comentará en la sección del cursillo de lenguaje máquina cómo poder modificar el programa (que por supuesto está escrito en código máquina), para poder adaptarlo a las peculiaridades de cada impresora (line feed automatico, etc.). Pero ¡¡tranquilos!! los programas aparecen listados también con un cargador en basic para que cualquiera pueda leerlos sin necesidad de saber código máquina, además, son muy cortos (no como la "cosa" que aparece en este número).

#### SORTEO DE DICIEMBRE DE COLABORADORES —PROGRAMAS Y MAGIA—

El sorteo de premios a los colaboradores cuyos esfuerzos han aparecido en los meses de julio a diciembre incluidos tendrá lugar, como dijimos, el 14 de diciembre en nuestra redacción. El único problema es que, debido a las fiestas de Navidad, este número lo cerramos antes de lo normal, hoy día 12, por lo cual no da tiempo a incluir los nombres de los ganadores en el mismo. La lista de los afortunados aparecerá en nuestro próximo número 12 del mes de febrero.

## BOLETIN COMPRA DEL NUMERO EXTRAORDINARIO

## 2.<sup>a</sup> EDICION

NOMBRE ..... TELEF. ....  
DIRECCION .....  
POBLACION ..... C. P. .... PROVINCIA .....

DESEO ..... EJEMPLAR (ES)

● PRECIO DE SUSCRIPCION 625 PESETAS ☐ ● LECTOR NO SUSCRIPTOR (25% dte) 935 PESETAS ☐  
● NO SOY SUSCRIPTOR PERO ADJUNTO BOLETIN DE SUSCRIPCION Y ME ACOJO AL PRECIO ESPECIAL  
PARA SUSCRIPTORES DE 625 PESETAS ☐

● Adjunto cheque por ..... Ptas. ● Envío giro nº ..... por ..... Ptas.  
Nº SUSCRIPTOR ..... (este número va en todas las etiquetas con la que se os envía la revista)



**Convocatoria del primer concurso nacional de programación en 80 columnas de Commodore-64 y 40-80 columnas del VIC-20 patrocinado por Ferre-Moret, S.A. Con la colaboración de Commodore World, más de 500.000 ptas. en premios.**

**BASES:**

- 1º Podrán participar todos los usuarios de COMMODORE-64 y VIC-20 que hayan obtenido la tarjeta de 80 columnas de FERRE-MORET, S.A.
- 2º Cada concursante podrá presentar, un programa realizado por él o por un equipo del que deberán indicar un titular.
- 3º Se enviará la explicación detallada del programa redactado en español, escrito a máquina. El programa se enviará en cassette o disco, adjuntando su correspondiente listado del programa.
- 4º Se enviarán por correo certificado a la siguiente dirección, FERRE-MORET, S.A. calle Buenos Aires, nº 30, 2º 3.ª de BARCELONA. Antes del 31 de enero de 1985.
- 5º Los programas no premiados podrán ser solicitados por sus propietarios en el plazo de treinta días después del fallo.
- 6º El resultado se comunicará, directamente a los ganadores y se publicará en la revista COMMODORE WORLD.

7º Los premios se entregarán durante la primera semana de marzo de 1985 coincidiendo con la entrega de premios del PRIMER CONCURSO NACIONAL ESCOLAR DE PROGRAMACION.

8º Los programas premiados quedarán en propiedad de FERRE-MORET, S.A.

9º Los programas que no obtengan premio, recibirán, en caso que así lo considere FERRE-MORET, S.A. ofertas de compra de sus programas.

- 10º Los premios ascenderán a 500.000 ptas. repartidos de la siguiente manera:
- 1º 250.000 Ptas.
  - 2º 150.000 Ptas.
  - 3º 100.000 Ptas.

11º Todos los participantes, que no hayan recibido premio u oferta de compra, podrán participar en el concurso de colaboradores que habitualmente COMMODORE WORLD viene realizando con un premio extra de FERRE-MORET, S.A. de 50.000 ptas.

12º Los trabajos deberán de ser inéditos y desarrollar temas de utilidad, programas profesionales, científicos, educativos, etcétera. FERRE-MORET, S.A. se reserva el derecho de modificar las bases de este certamen, dejarlo desierto o suprimirlo, por causa de fuerza mayor. En este caso la empresa se compromete a comunicarlo a través de COMMODORE WORLD.

## CONCURSOS PERMANENTES

### Colaboraciones de Programas y Magia

La verdad sea dicha: os estáis portando; las colaboraciones que nos están llegando son numerosas y de muy buena calidad. Por lo tanto, vamos a aclarar las condiciones de nuestros concursos permanentes.

- 1) Los sorteos se realizarán en los meses de junio y diciembre.
- 2) Se sortearán siempre seis premios, tres de 15.000, 10.000 y 5.000 pesetas en metálico de "Commodore World" y tres del mismo valor en material de Microelectrónica y Control, entre todas las colaboraciones publicadas.
- 3) Se sortearán siempre seis premios, tres de 3.000, 2.000 y 1.000 pesetas en metálico de "Commodore World" y tres del mismo valor en material de Microelectrónica y Control, entre todas las contribuciones publicadas en la sección de "Magia".
- 4) Adicionalmente, se sortearán diversos premios de cintas, juegos, etc. donados por firmas y distribuciones, entre todas las colaboraciones.
- 5) Los autores de las colaboraciones vendidas dentro de nuestro "Servicio de Cintas" percibirán el 20% del precio de la cinta.
- 6) Todas las colaboraciones deben venir escritas a máquina y los programas grabados en cinta (si es posible), o con el listado completo en impresora. Nuestros lectores más jóvenes pueden escribir a mano pero con letra muy clara.

7) Quedan automáticamente descalificadas tanto del sorteo como del "Servicio de Cintas" las colaboraciones que hayan sido enviadas a otras revistas.

8) Las colaboraciones se enviarán a "Commodore World", c/Barquillo, 21-3º Izda. 28004 Madrid.

### MAS VIAJES Y OTRO REGALO DE "IMPRESION"... A POR EL SUSCRIPTOR 8.000

Es evidente que a nuestros suscriptores les gusta viajar. De momento Joan Pujula visitará los Estados Unidos en julio y Juan Mejuto Iglesias irá a Inglaterra en el otoño. Queríamos sortear algo un poco especial ENTRE TODOS LOS SUSCRIPTORES cuando lleguemos al suscriptor número 8.000, y hemos optado por dos premios, uno concedido por Microelectrónica y Control y otro concedido por "Commodore World".

MEC regalará una impresora MPS 801 (o similar) y nosotros hemos escogido un surtido de viajes muy "chulos" para dos personas para que el ganador elija el que más le apetezca. La selección que hemos hecho es la siguiente:

Viajes de 4 días en hoteles de primera clase a uno de estos lugares: París, Lisboa, Atenas, Amsterdam, Niza, Viena.  
O viajes de una semana en hoteles de 4/5 estrellas: Canarias o Mallorca.

## INDICE DE ANUNCIANTES

	Pág.		Pág.
ALBAREDA	15	IDEALOGIC	30,31
ALPHA MUNDIAL GROUP,S.A.	34	ITAR	57
BASIC MICROORDENADORES	11	KENT ELECTRONIC	55
CASA DE SOFTWARE	6-7	MACHIN	17
CCC	39	MICROELECTRONICA Y CONTROL	46,47,76
CENTRO DE INFORMATICA	49	MICROINFORMATICA	49
COMMODORE WORLD (Distribución)	74	MICROSISTEMAS	49
COPERSA	21	MICROS GARDE	49
DINADATA	75	MICRO WORLD	49
DIRECTORIO	49	RADIO WATT	49
ELECTROAFICION	35	REM	49
ELECTRONICA SANDOVAL	49	RENT SOFT-64	53
FERRE MORET	25,42,43	SAKATI	27
FIRST	41	SOFTWARE ESPAÑA	2
ICR	56	TELE SANT JUST	45



casa de  
software s.a.

c/. aragón, 272, 8.º, 6.ª tel. 215 69 52 08007 barcelona

# Entra con nuestros programas en el mundo de la Música y el Diseño

## ■ LA OTRA MANERA DE ESCRIBIR



Escriba y dibuje sobre la misma hoja. Las posibilidades son ilimitadas... Informes, esquemas, ficheros gráficos, croquis con acotaciones, presupuestos, pequeños planos...

### CARACTERÍSTICAS:

- Hoja de 50 x 40 caracteres (400 x 320 pixels)
- Modo texto (puede crearse su propio alfabeto)
- Modo gráfico mediante joystick (4 velocidades)
- Modo robot. Volcado de gráficos predefinidos (32 x 24 pixels) sobre la hoja (128 gráficos por fichero).
- Grabación de la hoja en diskette o cassette
- Salida por impresora en alta resolución
- Versión cartucho

# SANDRA

## ■ EL INCREIBLE TECLADO MUSICAL



Transfórmese desde ahora en músico... creando, editando, componiendo y ejecutando sus propias piezas. El increíble teclado musical hace mucho más que convertir su commodore en un sintetizador.

**SIGHT & SOUND**  
Music Software, Inc.

### ARTICULO

### VERSION

### REFERENCIA

- INCREIBLE MUSICAL KEYBOARD	DISCO	C64D100
- INCREDIBLE MUSICAL KEYBOARD	CASSETTE	C64T100
- MUSIC PROCESSOR	DISCO	C64D101
- MUSIC PROCESSOR	CASSETTE	C64T101
- KAWASAKI SYNTHESIZER	DISCO	C64D102
- KAWASAKI SYNTHESIZER	CASSETTE	C64T102
- KAWASAKI RHYTHM ROCKER	DISCO	C64D103
- KAWASAKI RHYTHM ROCKER	CASSETTE	C64T103
- 3001 SOUND ODYSSEY	DISCO	C64D104
- 3001 SOUND ODYSSEY	CASSETTE	C64T104

## ORDENA TU ORDENADOR



Mueble sobremesa auxiliar

### DESEO RECIBIR INFORMACION:

Nombre y dirección:

---



---



---



Por Robert BAKER (RUN EE.UU.)

Traducido por Valerie SHANKS y adaptado por Diego ROMERO.

Creación/84

# Constructor de datos

**E**sta utilidad fue diseñada para los sistemas PET de Commodore pero el Listado 1 está adaptado para funcionar con el C-64 y el VIC-20.

El Programa lee un programa en lenguaje máquina de disco y crea un programa en Basic en disco con el mismo programa en lenguaje máquina convertido en sentencias Data. El programa también añade un bucle For...Next con los parámetros correctos para leer los datos e introducirlos en la memoria mediante unos Pokes. Esto te proporciona un método sencillo para convertir programas en lenguaje máquina en una forma que puedan ser mezclados con un programa en Basic.

Cuando se ejecuta el programa lo primero que hace es pedirte el nombre del fichero del programa en lenguaje máquina que será leído y convertido

(líneas 210-260). El programa en lenguaje máquina tiene que ser un fichero estándar y cargable como un programa (tipo PRG). No puede ser ningún tipo de fichero intermediario creado mediante un ensamblador. Si utilizas un ensamblador para crear el programa en lenguaje máquina, lo puedes cargar mediante el procedimiento normal y luego usar un monitor para salvarlo en disco como un fichero de programa. Una vez cargado como un fichero de programa, puede ser utilizado normalmente.

## Lo Que Hace

Una vez que el programa sabe el nombre de fichero del programa en lenguaje máquina, pide el nombre del nuevo programa en Basic que será

C-64 y VIC-20

creado (líneas 260-310). No debe de haber ningún fichero en el disco que lleve el mismo nombre. Si alguno de los ficheros no puede ser abierto, aparece un mensaje de error y el programa se da por terminado.

Observa que los comandos Open de las líneas 250 y 300 abren un fichero de programa (P) para leer y escribir, respectivamente. Esto no se puede realizar con el comando 4.0 DOpen en Basic. El comando standard Open debe ser utilizado tal y como se presenta.

Seguramente no encontrarás esto en la documentación de Commodore, pero los ficheros de programa pueden ser abiertos y utilizados igual que los ficheros secuenciales. Simplemente hay que tener en cuenta que los dos primeros bytes de un fichero especifican la dirección de carga, indicando dónde se cargará el programa en la memoria.

Una vez abiertos los ficheros necesarios, esta utilidad lee y saca en pantalla la dirección de carga del programa en lenguaje máquina (líneas 330-370). Cuando se abre el fichero para leerlo, los dos primeros bytes que se leen representan la dirección de carga en el formato 6502 (byte bajo/byte alto). Por lo tanto, la dirección se convierte en su valor decimal añadiendo el primer byte (el byte bajo) al segundo byte (el alto) multiplicado por 256.

A continuación, una dirección de carga de 1025 se coloca al principio del programa en Basic que se está creando con PRINT#2 en la línea 390. El valor 1025 es la dirección de carga estándar para todos los programas en Basic en los ordenadores Commodore. Si tienes un VIC-20 o un C-64, no hace falta modificar este valor dado que el cargador Basic vuelve a situar el programa cuando se carga en estos equipos. Por lo tanto, este valor es válido para todos los sistemas Commodore.

#### Entrando en el Bucle Principal

Una vez que el programa de utilidades sabe la dirección inicial; entra en el bucle principal, que lee un byte del programa en lenguaje máquina (línea 410), obtiene el valor decimal del byte (línea 420) y añade los datos a la línea del programa en Basic que se está construyendo en ese momento en L\$ (líneas 430-440). También se incrementa el contador de bytes para la longitud del programa en lenguaje máquina.

Se comprueba la longitud de la línea creada en el Programa en Basic en la línea 450 para ver si se pueden añadir más datos. De ser así, el programa vuelve a la línea 410 para leer el siguiente byte del programa en lenguaje máquina. De otra forma, se llama a la subrutina en la línea 580 para añadir esta línea al programa en Basic que se está salvando en disco. Después de escribir la línea en disco, el programa



vuelve a la línea 400 en vez de la línea 410 para colocar el "token" de los Datos (131) al principio de la línea en Basic.

La subrutina de las líneas 580-610 suma la longitud de la línea en L\$, más los cinco bytes adicionales para cada línea en Basic a un puntero en LK para calcular el enlace o dirección inicial de la siguiente línea en Basic. El valor del enlace de dos bytes se escribe en el fichero del programa en Basic seguido del número de línea de dos bytes de LN. A continuación, se presenta la línea de L\$, junto con un byte de cero como bandera de fin para indicar el final de la línea en Basic. Se utiliza otra subrutina en las líneas 620-630 para convertir la dirección de enlace y el número de la línea en Basic en el formato de direccionamiento de dos bytes 6502 y los escribe en el fichero.

#### Sólo un Byte más

Cuando el Programa de utilidades lee el último byte del programa en lenguaje máquina y detecta el final del fichero, cualquier dato restante será añadido al fichero en Basic (líneas 470-480). El valor de la variable de estado ST se salva en SS después de cada operación de lectura en el fichero en lenguaje máquina. Este valor será 64 cuando se llega al final del fichero.

Una vez escritos todos los datos en el fichero en Basic, se crea un bucle For...Next en L\$, insertando la longitud del programa en lenguaje máquina como el número de cuenta del bucle y la lectura de la dirección de carga será utilizada como la dirección del Poke inicial (líneas 500-540).

A continuación se escribe la línea del programa en el fichero en Basic con un enlace de cero (dos bytes, ambos de cero) para indicar el final del programa creado en Basic.

Se ha incluido un pequeño ejemplo del tipo de programa que crea el programa de utilidades. Observa que los números de línea del programa en Basic empiezan en la 10 y se incrementan de 10 en 10. Si quieres modificar el número para la línea inicial, vuelve a definir el valor de LN en la línea 180 de la forma que desees. Del mismo modo, si cambias el valor de LI en la línea 190 se modificará el incremento entre los números de línea.

El programa crea líneas en Basic limitadas a 78 caracteres o menos cuando salen en pantalla. Esto te permite utilizar el editor de pantalla para modificar las líneas si es necesario. Si quieres un resultado más compacto y no te importa no poder editar las líneas, puedes modificar el valor de prueba en la línea 450 de 65 a 250. El valor mayor hará que el programa de utilidades cree líneas de la máxima longitud que permite el Basic. Las líneas pueden salir en pantalla pero no pueden ser editadas.

### Cuando el Programa de utilidades

lee el último byte

del programa

en lenguaje máquina

y detecta el final

del fichero,

cualquier dato

restante

será añadido

al fichero

en Basic.





```

10 REM *****
20 REM
30 REM CONSTRUCTOR DE DATOS
40 REM
50 REM POR: ROBERT BAKER
60 REM
70 REM ADAPTADO POR D. ROMERO
80 REM
90 REM *****
100 :
170 :
180 LN=10:REM NUMERO DE LA PRIMERA L
  INEA BASIC
190 LI=10:REM INCREMENTO DE LINEAS
200 :
210 PRINT"[CLR]EL[SPC]PROGRAMA[SPC]D
  E[SPC]LENGUAJE[SPC]MAQUINA"
220 PRINT"[A]SPC]CONVERTIR[SPC]ES[SPC]
  -":PRINT
230 INPUT F$
240 OPEN15,8,15
250 OPEN1,8,5,"0":"+LEFT$(F$,16)+"",P,
  R"
260 INPUT#15,EN,EM$:IFEN<>0THENPRINT
  "ERROR[SPC]EN[SPC]DISCO-";EN,EM$:GOT
  0650
270 PRINT:PRINT"EL[SPC]PROGRAMA[SPC]
  GENERADO[SPC]SE[SPC]LLAMARA-":PRINT
290 INPUTF$
300 OPEN2,8,6,"0":"+LEFT$(F$,16)+"",P,
  W"

```

```

310 INPUT#15,EN,EM$:IFEN<>0THENPRINT
  "ERROR[SPC]EN[SPC]DISCO-";EN,EM$:GOT
  0650
320 PRINT:PRINT"OK, [SPC]CREANDO[SPC]
  EL[SPC]NUEVO[SPC]PROGRAMA...":PRINT
330 GET#1,C$:IFSTC<>0THEN#640
340 AD=0:IFC<>0"THENAD=ASC(C$)
350 GET#1,C$:IFSTC<>0THEN#640
360 C=0:IFC<>0"THENC=ASC(C$)
370 AD=AD+(256*C):PRINT"DIRECCION[SPC]
  DE[SPC]COMIENZO[SPC]=";AD:PRINT
380 LK=1025:NB=0
390 PRINT#2,CHR$(1);CHR$(4);
400 L$=CHR$(131):REM TOKEN DE 'DATA'

```

```

410 GET#1,C$:SS=ST:IFSS<>0THEN#470
420 C=0:IF C<>0"THEN C=ASC(C$)
430 IFLEN(L$)>1THENL$=L$+" "
440 L$=L$+MID$(STR$(C),2):NB=NB+1
450 IFLEN(L$)>65THEN#410
460 GOSUB580:GOTO400
470 IFSS<>64THEN#640
480 IFLEN(L$)>1THEN#GOSUB580
490 IFNB=0THEN#650
500 REM LAS SIGUIENTES LINEAS CREAM
  UNA LINEA BASIC
510 REM FOR X=0 TO...:READ C:POKE..
  ,X,C:NEXT
520 L$=CHR$(129)+"X"+CHR$(178)+"0"+C
  HR$(164)
530 L$=L$+MID$(STR$(NB-1),2)+"":+CHR
  $(135)+"C":
540 L$=L$+CHR$(151)+MID$(STR$(AD),2)
  +CHR$(170)+"X,C":+CHR$(130)
550 GOSUB580:PRINT#2,CHR$(0);CHR$(0)
  ;
560 PRINT"LONGITUD[SPC]=";NB+1;"BYTE
  S":PRINT
570 PRINT"CONVERSION[SPC]TERMINADA":
  GOT0650
580 L=LEN(L$):LK=LK+5+L:X=LK:GOSUB62
  0
590 X=LN:GOSUB620:LN=LN+LI
600 FORX=1TOL:PRINT#2,MID$(L$,X,1):
  NEXT
610 PRINT#2,CHR$(0);:RETURN
620 X1=INT(X/256):X2=X-(X1*256)
630 PRINT#2,CHR$(X2);CHR$(X1);:RETUR
  N
640 PRINT:PRINT"ERROR[SPC]EN[SPC]DIS
  CO, [SPC]PROGRAMA[SPC]INTERRUMPIDO"
650 CLOSE1:CLOSE2:CLOSE15

```

## Clave para interpretar los listados

Todos los listados que se publican en esta Revista han sido ejecutados en el modelo correspondiente de la gama de ordenadores COMMODORE. Para facilitar la edición de los mismos en la Revista y para mejorar su legibilidad por parte del usuario, se les ha sometido a ciertas modificaciones mediante un programa escrito especialmente para ello. Para los programas destinados a los ordenadores VIC-20 y COMMODORE 64, en los que se usan frecuentemente las posibilidades gráficas que aparecen del teclado, se han sustituido los símbolos gráficos que aparecen normalmente en los listados por una serie de letras entre corchetes [ ] que indican la secuencia de teclas que se deben pulsar para obtener el carácter deseado. A continuación se da una tabla para aclarar la interpretación de las indicaciones entre corchetes:

[CRSRD]= Tecla cursor hacia abajo (sin SHIFT)  
 [CRSRU]= Tecla cursor hacia arriba (sin SHIFT)  
 [CRSRR]= Tecla cursor a la derecha (sin SHIFT)  
 [CRSRL]= Tecla cursor a la izquierda (sin SHIFT)

[HOME]= Tecla CLR/HOME (sin SHIFT)

[CLR]= Tecla CLR/HOME (con SHIFT)

Las indicaciones [BLK] a [YEL] corresponden a la pulsación de las teclas de 1 a 8 junto a la tecla CTRL. Lo mismo sucede con [RVSON] y [RVSOFF] respecto a la tecla CTRL y las teclas 9 y 10.

El resto de las indicaciones constan de la parte COMM o SHIF seguidas de una letra, número o símbolo —por ejemplo [COMM+] o [SHIFA]—. Esto indica que para obtener el gráfico necesario en el programa deben pulsarse simultáneamente las teclas COMMODORE (la que lleva el logotipo) o una de SHIFT y la tecla indicada por la letra, el número o el símbolo, en el ejemplo anterior: COMMODORE y + o SHIFT y A, respectivamente.

En los signos gráficos además se cuenta el número de veces que aparece. Por ejemplo, [7 CRSRR] equivale a 7 pulsaciones de la tecla cursor a la derecha y [3 SPC] tres pulsaciones de la barra espaciadora.

# algunos de nuestros programas para los ordenadores personales

**commodore**



25.000pts.



15.000pts.



8.000pts.



15.000pts.



5.000pts.



10.000pts.



10.000pts.



15.000pts.



5.000pts.



10.000pts.



10.000pts.

PERMANEZCA ATENTOS  
A NUESTRAS PANTALLAS

**B.M.**

**BASIC MICRO-ORDENADORES, S.A.**

AVD. CESAR AUGUSTO,72

Tifs. 23 56 82 y 22 65 44

50003-ZARAGOZA

## Este instrumento importante para la programación del lenguaje de máquina es la clave para examinar las rutinas internas ROM del VIC-20 y el C-64.

El VIC-20 y el C-64 resultan ser muy eficaces para programar la animación, dadas sus maravillosas capacidades para el sonido, el color y gráficas. Todas estas características pueden utilizarse en BASIC, el lenguaje residente del VIC-20 y del C-64. Pero para adquirir una mayor velocidad, es mejor realizar la animación y otras aplicaciones importantes en el lenguaje máquina. Este programa está hecho para el VIC-20 pero también funciona en el C-64 modificando la siguiente línea:

```
100 DIMB$(15), A$(7): POKE 53281, 1: LBS$= "[19 SPC]": GOTO 700.
```

Esto resulta algo problemático para el programador no profesional, dado que los Commodore no disponen de un monitor interno para el lenguaje máquina. El hecho de que resulta difícil encontrar una buena documentación sobre la estructura interna del VIC-20 y el C-64 y una descripción pormenorizada de las rutinas ROM complica más el problema.

El programa presentado a continuación es un instrumento de lenguaje máquina que genera vucelos hexadecimales, vucelos de ASCII y desensamblajes; además proporciona varias funcio-



Por Thomas HENRY

Traducido por Valerie SHANKS, y adaptado por Diego ROMERO.

# Descubrir el secreto de la estru

nes adicionales también importantes para la programación del lenguaje máquina. Con el uso de este programa el usuario puede examinar las rutinas internas del ordenador y así adquirir unos conocimientos de primera mano sobre la estructura de la máquina ¡En cierto modo, el usuario está escribiendo su propia documentación!

### Sobre el Programa

Estudia el Listado 1 para familiarizarte con la estructura del Instrumento del Lenguaje Máquina. El programa lleva mucha documentación sobre las sentencias REM; en consecuencia, poco hay que decir sobre cualquier sección en especial.

Las líneas 110-140 forman una rutina de conversión de decimal en hexadecimal. Esta rutina se utiliza en muchas partes del programa —principalmente porque la función "PEEK" utilizada en las subrutinas de desensamblaje y vucelo hexadecimal proporciona un

## Todo el contenido de la memoria

se imprime en hexadecimal.

Además,

los equivalentes ASCII

se imprimen en rojo

en el lado

derecho de la pantalla.

Esto resulta

conveniente

para la búsqueda

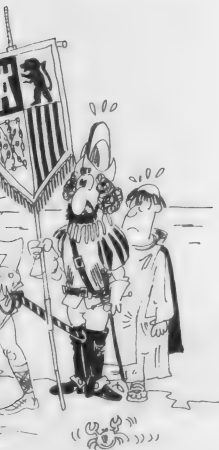
de mensajes de código.

resultado decimal. Se introduce la subrutina con D igual al número decimal (entre 0 y 65355) y se convierte en D\$ igual al equivalente hexadecimal. Observa el uso de la función CHR\$ para la generación de los dígitos correctos (0-9 y A-F).

La subrutina presentada en las líneas 150-210 realiza la función contraria: un número hexadecimal se convierte en el equivalente decimal. La rutina comienza con E\$ igual al número hexadecimal de cuatro dígitos y termina con E igual al equivalente decimal. En este caso, la función ASC se utiliza para obtener el número correcto.

Además, la subrutina lleva incorporada una función para detectar errores. Si la entrada hexadecimal contiene algún dígito no hexadecimal, o si su longitud no es exactamente de cuatro dígitos, aparece un mensaje de error y el usuario retorna al bucle principal.

La rutina de una sola línea en la línea 220 simplemente introduce un número hexadecimal. Se genera un campo



hacer la conversión a mano, así que deía que lo haga el ordenador). Lo que hace la rutina realmente es introducir un número y enviarlo a la subrutina en la línea 110.

Como la conversión de hexadecimal a decimal también resulta laboriosa, conviene automatizar las líneas 420-450. De nuevo se recurre a la rutina de utilidad (en la línea 150) para realizar la tarea.

Las líneas 480-540 realizan las rutinas de vuelco hexadecimal y vuelco ASCII. Se representan 20 líneas de vuelco hexadecimal en la pantalla, cada línea teniendo cuatro entradas. Esto significa que las direcciones saltan en grupos de cuatro, lo cual resulta una cifra conveniente para trabajar en hexadecimal.

Todo el contenido de la memoria se imprime en hexadecimal. Además, los equivalentes ASCII se imprimen en rojo en el lado derecho de la pantalla. Esto resulta conveniente para la búsqueda de mensajes de código.

El desensamblaje se incluye en las líneas 550-640. La mayoría de las rutinas utilizadas por el desensamblaje ha sido presentada anteriormente.

Sin embargo, observa que la línea 600 se ocupa de un caso de "código malo", imprimiendo tres interrogaciones consecutivas. La línea 630 llama a la rutina que corresponde a la modalidad de direccionamiento deseada; de esta forma la rutina genera la puntuación adecuada. La variable P sigue la pista de la dirección actual, mientras que CS es igual al mnemónico (p. ej. LDA, STX,

PLA). Más adelante hablaremos de cómo funciona el desensamblaje.

Las líneas 650-690 comprenden la rutina de nivel superior. Las teclas de función programables del ordenador se utilizan para llamar a las distintas características. Por ejemplo, la tecla F1 inicia el desensamblaje y la tecla F5 inicia el vuelco hexadecimal. Las teclas de función generan un código del ASCII de 133 a 137 y las líneas 680 y 690 buscan dicho código.

#### Estructura y Tablas

La última parte del programa consiste en la estructuración y las tablas. Se utiliza un "array" de variables de "strings" para almacenar el mnemónico necesario. Quizá este tipo de variable resulte un poco más lenta que las demás, pero tiene la ventaja de ocupar relativamente poco espacio. Esto es importante, ya que el VIC-20 estandar sólo dispone de 3.5 K de RAM utilizable. El "array" AS() contiene los 57 mnemónicos mientras que el "array" BS() contiene la información de direccionamiento.

Pondremos un ejemplo para explicar cómo estos "arrays" generan el desensamblaje correcto. Supongamos que un carácter en la memoria resulta ser igual al hexadecimal 25. El dígito más importante (el 2 en este caso) llama al "string" BS(2). Ahora, mirando la línea 590, el dígito menos importante (5) se multiplica por 3 y se incrementa por 1 para dar el número 16. WS se establece

## La estructura interior de los Commodore

inverso de cuatro bloques de longitud para indicar al usuario la longitud deseada del número de entrada.

Las líneas 230-360 contienen las subrutinas utilizadas por el desensamblaje. El cometido de estas rutinas es el de generar la "puntuación" adecuada para el campo operando (las comas, signos de dólar, paréntesis, etc.). Dado que el desensamblaje utiliza el mnemónico estandar 6502, cuesta bastante trabajo generar la puntuación más adecuada. Observa que la variable F es de 0, 1 ó 2, según sea la longitud de 0, 1 ó 2 caracteres. El "string" AS(0) contiene casi toda la puntuación necesaria, y las secciones deseadas se separan en la línea 250.

Las líneas 270-300 convierten las direcciones relativas en direcciones absolutas al decidir si la bifurcación es regresiva o anticipada, y si está restando o sumando.

La rutina en las líneas 370-410 te proporciona una conversión de decimal a hexadecimal. (Resulta muy laborioso

***Se utiliza un "array" de variables de "strings" para almacenar el mnemónico necesario.***

***Quizá este tipo de variable resulte un poco más lenta que las demás, pero tiene la ventaja de ocupar relativamente poco espacio.***

igual a MDS(BS(2),16,3), que resulta ser 12C. Este código toma el mnemónico correcto (representado por CS en la línea 620) a través del segundo mnemónico (AND en este ejemplo) en el "string" AS(1) — lo cual explica el 12 en el "string" 12C.

Se encuentra la modalidad de direccionamiento correcta al tomar ASC ("C"), es decir 67, y restarle 64, dejando 3. Por lo tanto, la subrutina 330 se llama en la línea 630 y se utiliza para generar la puntuación. El código lleva una dirección de página cero en este ejemplo.

Aunque esta rutina parezca un poco larga, en la práctica el programa funciona bastante de prisa. El desensamblaje de 20 líneas tarda seis segundos.

#### Introducir el Programa

Antes de introducir el Listado 1, ten en cuenta que aunque se vean muchas sentencias REM, no deben de introducirse en el ordenador. Esto sería imposible dada la memoria limitada del VIC-

20. Para que sea más fácil encontrar las sentencias que se tienen que introducir, observa que el programa empieza en la línea 100 y se va aumentando por decenas. Todas las sentencias REM se ubican en otros números. Así que comienza a partir de 100 e introduce todas las líneas que terminen en 0.

Los controles de cursor se utilizan de vez en cuando en las sentencias PRINT y ENTER. Dado que la impresora utilizada para generar este listado no imprime los símbolos de control del cursor (sólo las impresoras Commodore lo hacen), hay que emplear otros medios. Se utilizan las abreviaturas entre paréntesis para indicar los controles del cursor, y los REM en las líneas 31-51 indican cómo deben ser interpretados.

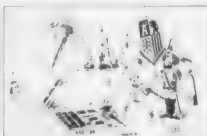
El programa es de fácil introducción, pero hay que tener cuidado que se mecanografien los espacios exactamente igual que en las distintas sentencias Print.

Después de entrar el programa, es aconsejable salvarlo. Después de comprobar que no hay errores de mecanografía y después de probar el programa, se debe hacer una copia de seguridad.

## Utilizar el Programa

Sólo se utilizan las teclas de función [F1] - [F7], para operar el programa.

Primero se carga y se ejecuta el programa. Después de que salga la pantalla con el título, el ordenador espera los mandatos:



• [F1] - Esta tecla inicia el desensamblaje. Después de pulsar la tecla, el ordenador te pedirá una dirección de comienzo. Teclea un número hexadecimal de cuatro dígitos y pulsa la tecla return. Así se inicia el desensamblaje. Si le das una dirección de comienzo errónea, recibirá un mensaje de error.

• [F3] - Esta tecla continúa el proceso de desensamblaje a partir del punto en que tú lo dejaste. Se desensamblan unas 20 líneas adicionales.

• [F5] - Esta tecla inicia el vuelo hexadecimal. Primero se solicitará una dirección de comienzo. Teclea un número hexadecimal de cuatro dígitos y pulsa la tecla return; así se inicia el vuelo hexadecimal. La columna de la izquierda representará la dirección de memoria en azul. Las cuatro columnas siguientes, también en azul, representarán el contenido de la memoria. Las últimas cuatro columnas, en rojo, representarán la interpretación ASCII del contenido de la memoria.

• [F7] - Esta tecla continúa el vuelo hexadecimal a partir del punto en que lo deja el usuario.

• [F2] - Esta tecla permite la conversión de un número hexadecimal en un número decimal. Después de pulsar la tecla [F2] el ordenador solicita un número hexadecimal. Teclea un número de cuatro dígitos y pulsa return. Se representará el equivalente hexadecimal. Si se teclea un número hexadecimal erróneo, se recibirá un mensaje de error.

• [F4] - Esta tecla convierte un número decimal en un número hexadecimal. Después de pulsar la tecla, se solicita un número decimal. Teclea el número y pulsa return. Se representará el equivalente hexadecimal. Ten en cuenta que la entrada tiene que ser entre 0 y 65535 ó, en caso contrario, se representará un mensaje de error.

Para que no tengas que aprender todo de memoria, puedes utilizar una plantilla de cartón para indicar con detalle los usos de las teclas de función. Esta plantilla se demuestra en la Figura 1. Se recorta la plantilla y se pega a un trozo de cartón. Se recorta y se tira la zona del centro. Así la plantilla se acopla alrededor de las teclas de función.

El Instrumento de Lenguaje Máquina resultará útil para entender la estructura interna de ordenador. Mediante el uso del desensamblaje, las rutinas pueden ser interpretadas y se puede descubrir el tipo de estructura necesaria para que éstas puedan ser introducidas. El vuelo ASCII resultará útil tanto para encontrar las palabras clave Basic en la memoria como para los mensajes de error.

## VIC-20

```

10 REM*****
*****
11 REM*
*
12 REM* VIC-20 MACHINE LANGUAGE TO
OL *
13 REM*
*
14 REM* THOMAS HENRY
*
15 REM* TRANSOMIC LABORATORIES
*
16 REM* 245 MORTON STREET
*
17 REM* MINNETONKA, MN 55001
*
18 REM*
*
19 REM*
*
20 REM*
*
21 REM*
*
22 REM*****
*****
23 REM
24 REM
25 REM
26 REM*****
*****

```

```

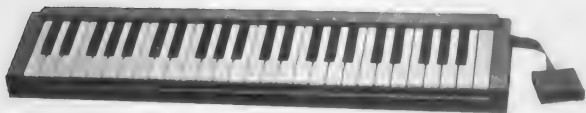
31 REM*
*
32 REM* PRUTAS PARA LA INTERPRETHC
ION *
33 REM* DEL LISTADO
*
34 REM*
*
35 REM* SHIFT+CLR [CLR]
*
36 REM* HOME CURSOR [HOME]
*
37 REM* CURSOR ARRIBA [CURSU]
*
38 REM* CURSOR ABAJO [CURSD]
*
39 REM* CURSOR IZQUIERDA [CURSL]
*
40 REM* CURSOR DERECHA [CURSR]
*
41 REM* CTRL+RVS ON [RVSUN]
*
42 REM* CTRL+RVS OFF [RVSOFF]
*
43 REM* CTRL+7 [BLU]
*
44 REM* CTRL+3 [RED]
*
45 REM*
*
46 REM* LOS SÍMBOLOS ENTRE CORCHET
ES *

```



# MUSIC 64

## PERSONAL COMPUTER MUSIC



### MUSICA CON COMPUTADOR con el Teclado "MUSIC 64"

El teclado "MUSIC 64" se conecta directamente al COMMODORE 64.

Se puede emplear como sintetizador monofónico o bien como teclado de órgano polifónico. Para ello se necesita lo siguiente:

- Un COMMODORE 64.
- Un video monitor o un receptor de televisión.
- Y una unidad FLOPPY DISK o de cassettes.

ALBAREDA le proporciona a usted:  
● Un teclado de 4 octavas de Do a Do.  
● Un adaptador para realizar el empalme del teclado con el computador.  
● El software necesario.

### EL SINTETIZADOR MONOFONICO

La relación de los instrumentos de los cuales se dispone, está indicada en el video monitor y son los siguientes:

- |                           |                            |
|---------------------------|----------------------------|
| 0 Trompeta                | 7 Instrumentos de          |
| 1 Brass                   | cuerda (cuerdas)           |
| 2 Clarinete               | 8 Piano                    |
| 3 Campanas                | 9 Organo eléctrico I       |
| 4 Flauta                  | A Organo eléctrico II      |
| 5 Guitarra                | B Acordeón                 |
| 6 Wha-Bras<br>(cobreswha) | C Random<br>(sintetizador) |

### Modificaciones paramétricas

F1 para alcanzar el parámetro siguiente.  
F3 para regresar al parámetro precedente.  
F5 para aumentar el valor del parámetro elegido.

F7 para disminuir el valor del parámetro elegido.

### EL TECLADO DEL ORGANO POLIFONO

La relación de los instrumentos de los cuales se dispone, está indicada en el video

monitor y son los siguientes:

- |                      |          |
|----------------------|----------|
| 1 Spinete            | 4 Flauta |
| 2 Acordeón           | 5 Banjo  |
| 3 Campanas tubulares |          |

### Modificaciones paramétricas

- |             |                          |
|-------------|--------------------------|
| F1 Ataque   | F2 Volumen               |
| F3 Declive  | F4 Traslado              |
| F5 Sostener | F6 Forma de la onda      |
| F7 Soltar   | F8 Ciclo de la operación |

Deseo recibir información completa del MUSIC-64 sin ningún compromiso por mi parte:

Nombre .....

Dirección .....

Teléf.: ..... Ciudad .....

C.P. .... Provincia .....

Soy distribuidor... ☐

Soy particular... ☐

(\*) Marcar con un asterisco lo que interese.

**Albareda**

INSTRUMENTOS  
Y ACCESORIOS MUSICALES

C/. Carmen, 19  
TARREGA (Lérida) - Teléfonos (973) 31 04 02 - 31 23 51





# Castellanizar el Commodore-64

**D**el **abecedario**: Siguiendo a la Real Academia Española en su opúsculo que recoge las Nuevas Normas de Prosodia y Ortografía (R.A.E., **Ortografía**, Madrid, Imprenta Aguirre, 1974<sup>2</sup>), puede observarse la carencia de la letra "Ñ", en su versión mayúscula y minúscula que, a partir de nuestro programa, se obtienen pulsando las teclas siguientes:

$$\begin{array}{ccc} \tilde{N} & \longrightarrow & \boxed{C=} \\ \tilde{M} & \longrightarrow & \boxed{C=} \end{array} \quad \begin{array}{c} y \\ x \end{array} \quad \begin{array}{c} \boxed{M} \\ \boxed{N} \end{array}$$

Aunque no es propio del abecedario castellano actual, nos ha parecido interesante añadir el carácter "ç" con el que nuestro ordenador queda "catalanizado" y preparado también para algún uso histórico de nuestro idioma. Se obtiene con  $\boxed{C} = \boxed{v} \boxed{C}$

### De los signos de puntuación y notas auxiliares

En castellano, la existencia en algunos casos de una tilde sobre la vocal, además de responder a unas normas ortográficas bien claras y precisas (con cit., cap. III), que tienden a evitar anfibologías ortográficas (caso de **deposító**, **deposító**, **deposító**, por ejemplo), su uso representa al acento de intensidad de la pronunciación y responde perfectamente al "desideratum" de cualquier sistema de escritura: correlato unívoco entre sonido y grafía. Para evitar un programa complejo que supliera la imposibilidad del ordenador (salvo con el uso de "sprites") de sobreimpresionar caracteres (como la máquina de escribir tiene para el caso de " " ), se ha diseñado un nuevo juego de caracteres vocálicos que, ajustándose estrictamente a las demás formas tipográficas del COM-MODORE, incorporan la tilde. Se obtienen pulsando:

[C=] y la vocal correspondiente.

Admitiendo una costumbre tipográfica que, poco a poco, se va implantando, se excluyen las vocales mayúsculas con acento, pese a que en las Normas de la RAE se insiste expresamente en que "el empleo de mayúsculas no exige de poner tilde..." (párrafo 6, apdo. 15º); sin embargo, su inclusión convertiría el teclado del COMMODORE en un alarde nemotécnico para el programador al tenerse que asignar los caracteres a otras teclas distintas a las vocálicas (por tener éstas ya ocupadas sus tres posibilidades: mayúscula, minúscula y minúscula con tilde). Este es el caso de la forma "ü" (en *desague*, por ejemplo) que hemos situado en lugar de [G] (pulsación directa) por imposibilidad de incorporarlo en su letra correspondiente. No incluimos la diéresis ("...") en

El programa que proponemos trata de incorporar al **COMMODORE 64** algunos caracteres, signos de puntuación y notas auxiliares atípicos del inglés pero que, en castellano, son imprescindibles en unos casos y propios de nuestra idiosincrasia idiomática en otros.

Por Angel ROMERO BELMONTE  
Profesor de Lengua y Literatura Españolas  
Instituto de Bachillerato  
de Fuengirola  
(Lector)



C O M E N T A R I O S	D E L A U T O R
---	--------------------------------------

Soy profesor de Lengua y Literatura Españolas en el Instituto de Bachillerato de esta población. Aficionado a la informática, he seguido su revista COMMODORE WORLD desde el comienzo, ello me lleva a enviarles el adjunto artículo para que, si les parece, sea publicado como colaboración desinteresada.

Creo que en nuestro mundillo de "comodorianos" puede resultar interesante el doble objetivo del artículo: cuidar nuestro idioma y "castellanizar" el teclado del ordenador.

## EN DEFENSA DEL IDIOMA

No se trata de defender aquí, con un mal entendido purismo academicista, la limpieza de nuestra lengua; todo lo contrario, las lenguas no son productos "incontaminados" sino que, o se hallan genéticamente emparentadas, o se enriquecen por "contagio lingüístico" siguiendo una especie de vaivén histórico, de predominio de unas sobre otras (piénsese cómo, en el terreno concreto de la ciencia, el galicismo fue el dominante en los siglos XVI-XVII y el casticismo dejó su impronta durante los siglos XVIII-XIX). Estos hechos, inevitables, por otra parte, no tienen nada de censurable. Son formas de enriquecimiento cuando los "préstamos léxicos" no vienen a superponerse a denominaciones ya existentes en el

## -Commodore World Enero 1985/19



Línea 4 → Copia en RAM la información sobre los caracteres de origen que posee la ROM y pone el teclado para comenzar en minúsculas (como la posición inicial de una máquina de escribir).

Línea 5 → Vuelve a conectar los registros de E/S.

Línea 6 → Vuelve a permitir el uso del teclado.

Línea 7 → Coloca el "puntero de caracteres" en la nueva dirección de la RAM a la que ha de dirigirse el microprocesador para encontrar la nueva información.

Si el programa se detuviese aquí, tendríamos un juego de caracteres y de gráficos sin ninguna diferencia con el original; pero vamos a crear nuestros propios caracteres. Tomemos el carácter "á" (definido en las líneas 12-13). Hemos de averiguar dos cosas: la dirección de pantalla a la que queremos asignar el carácter, y los datos, en decimal, que definen dicho carácter. Veamos lo primero.

Como se sabe, el C-64 asigna a las teclas 2 juegos de caracteres y 3 posibilidades en cada juego. En uno:

Pulsación directa → mayúsculas  
[SHIFT] y [Letra] → gráfico de la derecha.

[C=] y [Letra] gráfico de la izquierda.  
En el otro:

Pulsación directa → minúscula  
[SHIFT] y [Letra] → mayúscula  
[C=] y [Letra] → gráfico de la izquierda.

Esta última solución es la que hemos adoptado para nuestros propósitos. Tomemos la tecla "A". En pulsación directa nos da "a"; con [SHIFT], tenemos "A" y ahora queremos poner "á"

rancia de los mecanismos de creación castellanos (sirva de ejemplo el caso de la palabra **indexado**, del inglés "index", olvidando que el término lo tomó el inglés del nominativo latino, pero que la base patrimonial del castellano procede del acusativo, con lo que el término, sin necesidad de recurrir a otras lenguas, ha de ser **indicado**, del lat. **indice (m)**, como lo es el cultismo **índice**. Los ejemplos podrían multiplicarse abundantemente y no es este el propósito.

Hay que empezar también, a apreciar, como valor técnico importante, a aquellos ordenadores cuyo teclado viene de origen "españolizado" (que sepamos, algunos microprocesadores ya lo traen incorporado), presionando así a los fabricantes e importadores a plantearse seriamente la deficiencia.

Entre tanto ocupamos el lugar que nos corresponde en el mundo de la ciencia y de la técnica actual, "...la defensiva pura se estima como un medio provisional para recobrar la iniciativa que conduzca a la victoria..." (Vid, S. GILI GAYA, "El lenguaje de la ciencia y de la técnica", en **Presente y futuro de la lengua española**, Madrid, Instituto de Cultura Hispánica, II, 1964).

Para los que ya poseemos nuestro COMMODORE 64, proponemos las siguientes páginas que, en cierto modo, pueden paliar este problema.

Caracteres añadidos al Commodore 64

Carácter	Forma de obtenerlo	Código de pantalla	Posición de inicio en RAM
ñ	[C=] y [M]	103	13112
ñ	[C=] y [N]	106	13136
á	[C=] y [A]	112	13184
é	[C=] y [E]	113	13192
í	[C=] y [I]	98	13072
ó	[C=] y [O]	121	13256
ú	[C=] y [U]	120	13248
ü	[C=] y [U]	0	12288
ç	[C=] y [C]	124	13280
;	[C=] y [C]	31	12536
;	[C=] y [C]	30	12528
;	[C=] y [C]	105	13128
;	[C=] y [C]	95	13048
;	[C=] y [C]	100	13088
;	[C=] y [C]	122	13264
;	[C=] y [C]	104	13120

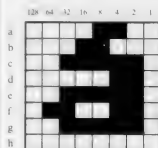
en lugar del gráfico de la izquierda. Si localizamos dicho gráfico en el **Manual del Usuario** (pp. 132-134) veremos que le corresponde la cifra 112 como "código de pantalla". Para calcular la dirección de memoria a partir de la cual se va a recoger la información sobre este carácter, aplicamos la siguiente fórmula:

$$[P=12288 + (C*8)]$$

en donde P = nº de posición de inicio en RAM (dato buscado) y C = Código de pantalla (112, en nuestro ejemplo). Obtendremos como dirección de memoria la 13184 que es nuestro primer dato.

Obtengamos el segundo dato, es decir, las cifras que recojan la información sobre la forma del carácter deseado. Sobre un cuadrado dividido en 64 partes, dibujamos nuestro carácter y obtenemos el núm. que corresponde a

cada línea. Así, por ejemplo, de la línea (a) hemos pintado los cuadrados con valores 8 y 4; de la línea (b), los valores 16 y 8, etc.,



- a) 8+4=12
- b) 16+8=24
- c) 32+16+8+4=60
- d) 4+2=6
- e) 32+16+8+4+2=62
- f) 64+32+8+4=102
- g) 32+16+8+4+2=62
- h) ningún cuadro=0

Por tanto los datos para definir a "a" son: 12, 24, 60, 6, 62, 102, 62, 0

Ahora se trata de aplicar la siguiente subrutina para situar estos datos en sus lugares de memoria correspondientes:

—FORB=P (posición de inicio) TOP+8 (última posición del carácter):

READC (datos decimales del carácter): POKEB, C: NEXT.

—DATA (8 núm. decimales del carácter)

Para nuestro ejemplo, tendríamos:  
12 FORB=13184 TO 13191: READC:  
POKEB, C: NEXT  
13 DATA 12, 24, 60, 6, 62, 102, 62, 0

Definidos todos los caracteres (16 en nuestro programa) y ordenadas las líneas en sucesión, tendríamos el listado nº 1.

Por Angel  
ROMERO  
BELMONTE,  
Profesor de Lengua  
y Literatura  
Españolas, Instituto  
de Bachillerato de  
Fuengirola (Lector).

```

0 PRINT"CLR":PRINT"[10CRSRD][11CRSRK]
ESPERE[SPC]UN[SPC]MOMENTO"
1 POKE52,48:POKE56,48:CLR
2 POKE5334,PEEK(56334)AND254
3 POKE1,PEEK(1)AND251
4 FORA=0TO2047:POKEA+12288,PEEK(A+55
296):NEXT
5 POKE1,PEEK(1)OR4
6 POKE5334,PEEK(56334)OR1
7 POKE53272,(PEEK(53272)AND240)+12
8 FORN=0TO15
9 READA,B
10 FORM=ATOB:READC:POKEM,C:NEXTM
11 NEXTN
12 DATA13112,13119,60,0,102,118,126,
110,102,0
13 DATA13136,13143,60,0,124,102,102,
102,102,0
14 DATA13184,13191,12,24,60,6,62,102,
62,0
15 DATA13192,13199,12,24,60,102,126,
96,60,0
16 DATA13072,13079,12,24,0,56,24,24,
60,0
17 DATA13256,13263,12,24,60,102,102,
102,60,0
18 DATA13248,13255,12,24,102,102,102,
102,62,0
19 DATA12288,12295,102,0,102,102,102,
102,62,0
20 DATA13280,13287,0,0,60,96,96,60,2
4,48
21 DATA12536,12543,24,0,0,24,24,24,2
4,0

```



```

22 DATA12528,12535,24,0,24,48,96,102
,60,0
23 DATA13128,13135,60,98,24,102,24,7
0,60,0
24 DATA13048,13055,8,26,62,28,8,0,0,
0
25 DATA13088,13095,56,108,108,108,56
,0,124,0
26 DATA13264,13271,56,12,60,108,60,0
,124,0
27 DATA13120,13127,102,102,102,0,0,0
,0,0
28 PRINT"CLR"

```

## Para los distribuidores de Commodore

ATENCIÓN A NUESTRA OFERTA

Cedemos a los Distribuidores de Commodore, parte de una larga lista de juegos de alta calidad, desconocidos en su mayoría en España, para el Commodore 64. En cinta y en disco. Desde 700 Ptas. en función de pedido.

También disponemos de aplicaciones de Gestión, en disco y cinta para todos los modelos de COMMODORE. Confeccionamos aplicaciones a medida y adaptamos los estándares a las necesidades específicas del cliente. Estamos especializados en gestión de empresa.

### GRAN NOVEDAD:

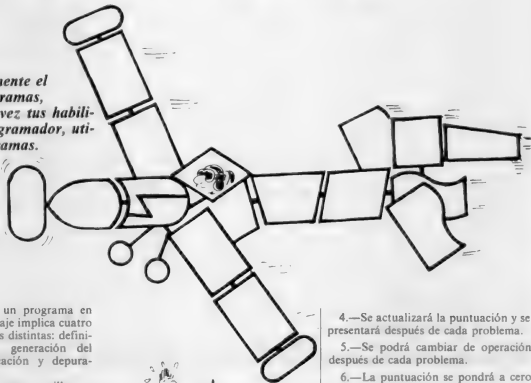
Aplicación de control de Almacén y Facturación directa. Controla 1.500 fichas de artículos sin cambiar de disco. Amplias estadísticas. Para Commodore 64 con unidad de monofloppy. Diseña la factura o se adapta a la imprenta.

**COPERSA:**

c/Fray Luis de León, 2. Teléfonos (983) 30 08 00 y 30 09 00 - 47002 Valladolid  
Delegación y Servicio Técnico de Commodore y Seiko.

# El arte de realizar organigramas

*Plasma gráficamente el fluir de tus programas, mejorando a la vez tus habilidades como programador, utilizando organigramas.*



**L**a escritura de un programa en cualquier lenguaje implica cuatro procesos o fases distintas: definición del programa, generación del organigrama, codificación y depuración.

Aunque en programas sencillos se puede evitar el realizar una o dos de las bases sobre el papel, siempre hay que realizarlas todas, aunque sólo sea mentalmente. Cada persona tiene su propia forma de trabajar; algunos son capaces de codificar directamente un programa de gran complejidad.

Sin embargo, la mayoría de los programadores tienen que realizar todas las fases descritas, incluso en los programas más sencillos. Los programas desarrollados ateniéndose a dichos pasos suelen ser, generalmente, de mejor calidad que aquellos que se codificaron directamente.

## Definición del Programa

Antes de escribir un programa, lo primero que debes tener claro es lo que quieres hacer. Esto puede parecer obvio pero, como saben los estudiantes de ciencias, el primer paso para resolver un problema es definirlo.

Supón que quieres escribir un programa para que un estudiante realice ejercicios aritméticos. No es suficiente saber que quieres escribir una lección de aritmética. Debes tener en mente un conjunto completo de objetivos y lo mejor para ello es escribirlos.

Tus objetivos deben estar bien detallados, pero no creas que no van a cambiar una vez escritos. Puedes, y debes revisarlos según vaya evolucionando el programa aunque deberías considerarlos inicialmente con cuidado para saber lo que realmente quieres hacer.

Por ejemplo, empecemos con los siguientes objetivos:

- 1.—Los problemas aritméticos se generarán aleatoriamente.
- 2.—El estudiante podrá seleccionar cualquiera de las cuatro operaciones aritméticas (suma, resta, multiplicación, y división).
- 3.—En cada una de las operaciones, el estudiante podrá seleccionar entre cuatro grados de dificultad.

4.—Se actualizará la puntuación y se presentará después de cada problema.

5.—Se podrá cambiar de operación después de cada problema.

6.—La puntuación se pondrá a cero tras cada cambio de operación.

7.—Se permitirá poner a cero la puntuación sin cambiar de operación.

8.—Cada vez que se resuelva un problema se indicará si la solución es correcta.

9.—Si la respuesta está equivocada se mostrará la solución correcta.

10.—Se generarán informes sobre los progresos del estudiante, basados en la puntuación y el nivel de dificultad.

Una vez definidos los objetivos del programa podemos pasar a la siguiente fase.

## Realización del Organigrama

Cualquier programa, no importa en qué lenguaje esté escrito (en código máquina o en un lenguaje de alto nivel como el BASIC), está formado por un conjunto de secuencias de operaciones y un conjunto de puntos de decisión desde los que se seleccionan las secuencias a ejecutar.

Mediante un organigrama (representación gráfica de dichas secuencias y decisiones) se puede visualizar fácilmente el flujo del programa. Si está bien hecho, el programador puede estructurar el programa de forma eficiente.



C-64 - VIC-20 SIN AMPLIAR

Mediante un organigrama el autor puede:

—Asegurarse de que se satisfacen todos los objetivos.

—Verificar que se han considerado y previsto todas las posibilidades.

—Identificar el código necesario en varios sitios dentro del programa, y segregarlo si procede en forma de subrutinas evitando, de esta forma, la redundancia de código.

Además, si se utilizan organigramas, se simplifica el resto de los procesos y el programa resultante es más profesional.

Los símbolos utilizados en los diagramas se atienen a ciertas normas, y si es preciso realizarlos con pulcritud, es posible conseguir plantillas especiales sin gran dificultad. Sin embargo, no es totalmente obligatorio tener una plantilla para realizar bien un organigrama, ni tampoco es necesario utilizar todos los símbolos que se han definido. De hecho, muchos programas, si no todos, pueden representarse adecuadamente con tan sólo tres símbolos.

En la Figura 1 se muestran los símbolos utilizados en los organigramas, tomados de una plantilla basada en la norma americana X3.5-1966. Entre ellos, los tres más utilizados son: decisión, proceso, y proceso predefinido y conector.

Los procesos predefinidos, normalmente llamados subrutinas, suelen distinguirse añadiendo una línea vertical u horizontal al símbolo. Esto se utiliza para indicar que dicho símbolo engloba muchos pasos de programas, posiblemente incluyendo operaciones y decisiones. Descubrirás que es muy útil usar estos procesos predefinidos en todos los casos salvo, quizás, en los más sencillos.

En la Figura 2 se muestra una variante del símbolo de decisión, que resulta muy útil cuando existen muchos puntos de salida (por ejemplo, la sentencia ON de BASIC). Esta situación puede representarse, también, por una secuencia de bloques de decisión única; pero esta variante es más compacta y simplifica la visualización del flujo del programa.

### Organigramas organizados sin fallos

Cuando se diseña un organigrama, se deben tener en cuenta 4 factores importantes.

En primer lugar, se debe suponer que el organigrama lo va a tener que interpretar otra persona y que tú no vas a estar allí para explicárselo. Tú mismo podrás apreciar los resultados de este supuesto, si uno o dos años después, decides modificar el programa para realizar una nueva aplicación. En ese momento, tú eres la otra persona y no hay nadie que pueda explicarte el organigrama.

En segundo lugar, no incluyas demasiados detalles. La idea es crear una imagen gráfica del programa. Si se inclu-

yen demasiados detalles, se distorsiona la imagen. Los detalles se colocarán en la siguiente fase.

En tercer lugar, cuando hayas concluido el organigrama, revisa los objetivos del programa fijados en la primera fase para asegurarte de que se cumplen. Este puede ser un buen momento para revisar los objetivos pero, asegúrate de que cualquier modificación que realices quede reflejada en el organigrama.

Finalmente, comprueba que se han considerado todas las posibilidades y se han previsto los flujos necesarios. Por ejemplo, ¿qué ocurriría si se realizara una entrada, por teclado, distinta de la esperada? o ¿qué ocurriría si se realizara una división por cero?

Durante la fase de realización del organigrama debes comenzar a elaborar una lista con los nombres de las variables que utilizarás en el programa. Durante esta fase, se suelen detectar las variables necesarias para su utilización como contadores de bucle, indicadores, para registrar las opciones seleccionadas por el operador y para almacenar los datos introducidos mediante el teclado. Los nombres de variables asignados en esta etapa deben utilizarse en el diagrama de flujo e incluirse en la lista de variables. Esto simplificará el proceso de codificación, al reducir el número de veces que deberás repasar la lista.

La creación de la lista de variables continuará en la fase de codificación, ya que, sin duda, necesitarás más variables, cuando llegues al último nivel de detalle.

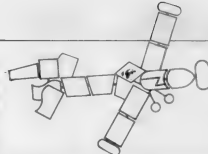
Esta lista te reportará muchas ventajas posteriormente. Eliminarás el tiempo que normalmente se pierde buscando en el código del programa si un nombre de variable ya ha sido utilizado, o volviendo a codificar partes en las que hayas utilizado nombres de variable previamente asignados.

La figura 3 muestra el diagrama de un programa que cumple los objetivos previamente definidos.

En este programa, como en la mayoría de los programas que forman bucles por naturaleza, la primera ejecución del bucle debe ser diferente del resto. La razón es que deben establecerse las condiciones iniciales. Estas se utilizarán para que el programa comience de forma ordenada.

En este organigrama, el primer bloque refleja el establecimiento de condiciones iniciales. Observa que el flujo del programa tiene en cuenta estas condiciones iniciales y no imprime los indicadores de función o de dificultad en esta primera pasada.

En este momento, se incluyen los nombres de variables que ya se conocen. Se identifican las subrutinas. De hecho, en este momento es posible, e incluso aconsejable, elegir los números de línea de comienzo de las subrutinas. Para el



DECISION	CINTA PERFORADA
PROCESO Y PROCESO PREDEFINIDO	CINTA MAGNETICA
CONECTOR	TERMINAL
OPERACION MANUAL	TARJETA PERFORADA
OPERACION AUXILIAR	DOCUMENTO
ENTRADA Y/O SALIDA	ENTRADA MANUAL
ALMACENAMIENTO FUERA DE LINEA	ALMACENAMIENTO EN LINEA
VISUALIZACION	DIRECCION DEL FLUJO
LINEAS DE COMUNICACIONES	

Fig. 1. Símbolos estándar de un diagrama de flujo.

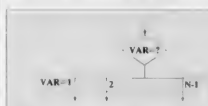
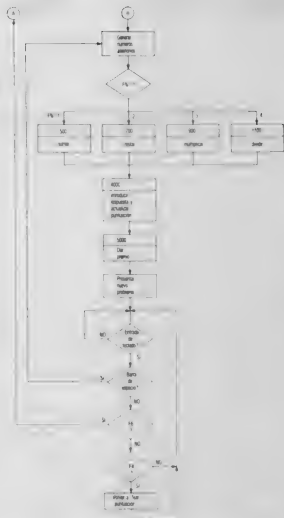


Fig. 2. Símbolo de decisión con salidas múltiples.

VIC-20 y el Commodore 64, no debe importarte utilizar números de línea altos ya que todos ocupan lo mismo en memoria (2 octetos).

La Guía de Referencia del Programador del VIC-20 tiene un error respecto a este punto. El único efecto que sobre la memoria puede tener el utilizar números de línea altos es el acortamiento de

Fig. 3 Diagrama de flujo de un programa de muestra



Otros puntos importantes acerca del organigrama de la Figura 3 son:

—El programa sólo genera un problema si se han seleccionado las opciones necesarias.

—No hay secuencias sin salida.

### Codificación

Cuando hayas escrito algunos pro-

También es importante documentar internamente el programa, utilizando sentencias de comentario (REM). Esto será especialmente útil si tienes que revisar tu programa dentro de mucho tiempo.

### Características Especiales

1. Un flujo lógico con números de línea incrementados regularmente.

Cuando el programa se ha codificado directamente, los números de línea no se incrementan regularmente debido a la inserción de objetivos descubiertos sobre la marcha, a la conversión de bloques redundantes en subrutinas o a la adición de sentencias para corregir problemas encontrados en la depuración.

2. Pocos GOTO. Es más fácil añadir nuevas líneas colocándolas fuera del flujo principal del programa y utilizando la sentencia GOTO para entrar y salir, que reenumerar todas las sentencias. Aquí se ha utilizado otra técnica para desarrollar el programa.

3. Los segmentos o módulos del programa están definidos con claridad.

Esto es importante cuando es deseable utilizar estos módulos en programas futuros. Si dispones de un amplio conjunto de programas que puedes utilizar en el desarrollo de otros nuevos, obtendrás notables ahorros de tiempo. Los programas que se codifican directamente no son realmente modulares. Esto dificulta tanto el reconocimiento de las líneas que forman los segmentos, como el traslado de éstos a otros programas.



# ¡INCREIBLES TARJETAS PARA COMMODORES 64 Y VIC-20 DE 80 COLUMNAS!



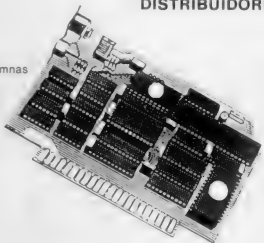
LA TARJETA DE 64K ES LA PERFECTA COMBINACION PARA LA TARJETA DE 40/80 COLUMNAS:

- Puede escoger entre 22/40/80 columnas y obtiene 31743 bytes libres!!!

- El software del EPROM de la tarjeta le permite utilizar todos los comandos de ficheros (OPEN, CLOSE, PRINT, INPUT, GET, SAVE and LOAD) con los 31232 Bytes de Memoria «oculta», lo que produce la sensación de que hay 8 cassettes super rápidos conectados a su Vic-20 Imagínese... menos de 1 segundo para guardar 28 K!!!

DISTRIBUIDORES PARA TODA ESPAÑA

- Tarjeta VIC-20 40/80 columnas
- Tarjeta COMMODORE 64 80 columnas
- Tarjeta VIC-20 Ampliación de memoria 64 K - RAM
- Slots de expansión para VIC-20 y 64 de 2 y 5 para conexión de tarjetas



PEDIDOS

**FERRE - MORET S.A.**

DISTRIBUIDORES Y PARTICULARES

C/Tusset, 8 - ENTRESUELO 2ª  
Teléf.: 218 02 93  
08006 BARCELONA

## ¡ATENCIÓN! SOFTWARE 80 COLUMNAS

### AGENDA 80 FMD

Sin duda la mejor Agenda Personal creada para el CBM-64, y a 80 Columnas.

Opciones: ALTAS, CONSULTAS (pantalla o impresora), MODIFICAR, BAJAS, LISTADOS (pantalla o impresora), ONOMASTICAS. Esta última opción le permite visualizar el calendario de un mes cualquiera de un año cualquiera. Le señala las fechas en que hay alguna onomástica y a continuación le indica el tipo y de quién es la onomástica. Al inicio del programa le indica si existe alguna onomástica en el día de hoy. Consultas por: Número, nombre, calle, población, Dto. postal, santo, y cumpleaños.

### BIBLIO 80 FMD

Archive su biblioteca y consulte luego por 5 conceptos diferentes: Número, Título, Autor, Clasificación y Grupo. Opciones Menú: ALTAS, CONSULTAS (pantalla o impresora), MODIFICAR, BAJAS, LISTADOS (pantalla o impresora). ¿Quiere saber los libros que tiene que tratan de Economía? Simplemente utilice la opción CONSULTAS y obtendrá un listado de sus libros de Economía.

### DISCO 80 FMD

Archive su Discoteca y consulte luego por 5 conceptos diferentes: Número, Título, Cantante/grupo, Editora y Clasificación. Las fichas le permiten especificar las canciones contenidas en el disco. La opción CONSULTAS, le permite conocer todos los discos que tienen un concepto en común. Por ejemplo, discos de los Beatles.

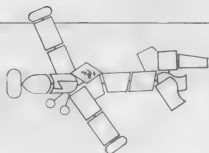
### REST 80 FMD

Sólo para gastrónomos. Consulta los Restaurantes por Número, Nombre, Población, Precio por Persona, Platos recomendados, Clasificación y Cocina. Las opciones del Menú son: ALTAS, CONSULTAS (pantalla o impresora), MODIFICAR, BAJAS, LISTADOS (pantalla o impresora). Ten tu propia guía de los restaurantes que has visitado.

### CINE 80 FMD

Sólo para cinéfilos. Ten tu propio archivo de películas y consulta por Número, Director, Productor, Fotografía, Música, Actores, Actrices, Clasificación y Grupo. Opciones del Menú: ALTAS, CONSULTAS (pantalla o impresora), MODIFICAR, BAJAS, LISTADOS (pantalla o impresora).

PROXIMAMENTE ESTOS MISMOS PROGRAMAS PARA CASSETTE Y 40 COLUMNAS.



4. Finalmente, la fase restante (depuración) es mucho más simple.

#### Depuración

La depuración del programa es sencilla si se siguen los pasos que hemos descrito.

En primer lugar, gracias a la labor previa de planificación, no habrá muchos fallos, si bien es cierto que la planificación no puede evitar errores mecanográficos.

En segundo lugar, la estructura lógica, la representación gráfica de dicha estructura y la lista de variables propician el seguimiento del flujo del programa y la detección de cualquier problema. Tendrás que utilizar toda tu

habilidad y experiencia en técnicas de depuración para resolver los problemas que puedan aparecer.

Las dos técnicas de depuración más útiles son el comando STOP y la instrucción GOSUB.

Para comprobar si se está ejecutando un segmento determinado o si los valores de las variables son los adecuados, se pueden colocar temporalmente comandos STOP a lo largo del programa. Cuando el programa se detiene a causa de un comando STOP es posible examinar el contenido de las variables, o cambiar sus valores utilizando comandos directos, y luego continuar la ejecución del programa. Esta ejecución no podrá continuar si se cambia una línea, o se añaden o borran líneas después de la parada.

Cuando conviene ejecutar una subrutina GOSUB para comprobar su funcionamiento, puede ser difícil o tedioso conseguir que el programa la llame. Este sería el caso si se intentara depurar la subrutina REWARD en el programa ejemplo. La siguiente inserción temporal en el programa evitaría esto:

1 M=1; F%=1; N%=4; YES=20; NO=0  
2 GOSUB 5000  
3 STOP

Ahora se puede ejecutar este programa temporalmente para comprobar que la subrutina funciona adecuadamente. Las condiciones de la línea 1 pueden cambiarse fácilmente para verificar la subrutina ante distintas hipótesis.

Cuando se hayan realizado todos estos pasos y tu programa esté acabado, le hayas puesto un nombre y lo hayas salvado, debes identificar la documentación producida durante el desarrollo con el nombre del programa y archivarla para su uso posterior.

#### ¿Agobiado?

Un programador sin experiencia puede sentirse agobiado al pensar que debe hacer todo lo que hemos explicado. Sin embargo, es difícil explicar con palabras las ventajas que reporta este método de programación. Cuando se escriben los programas de esta forma, son mejores y se hacen más rápidamente. Sólo la experiencia te convencerá de todo esto. ●

#### C-64

```
5 REM MATEMATICAS C-64
10 YES=0: NO=0: F%=0: N%=0
20 PRINTCHR$(147)"[2SP]MATEMATICAS[2SPC]
  :PRINT:PRINT:REM OPCIONES
30 PRINT"1.[SPC]ELIJE[SPC]UNA[SPC]OP
  ERACION[SPC]PULSANDO:"
40 PRINTTAB(4)"1[SPC]PARA[SPC]SUMAR"

50 PRINTTAB(4)"2[SPC]PARA[SPC]RESTAR
  "
60 PRINTTAB(4)"3[SPC]PARA[SPC]MULTI
  Plicar"
70 PRINTTAB(4)"4[SPC]PARA[SPC]DIVIDI
  R"
80 PRINT"2.[SPC]ELIJE[SPC]NIVEL[SPC]
  DE[SPC]DIFICULTAD[SPC]PULSANDO:"
90 PRINTTAB(4)"1[SPC]FACIL"
100 PRINTTAB(4)"2[SPC]F3[SPC]FACIL"
110 PRINTTAB(4)"3[SPC]F5[SPC]DIFICIL
  "
120 PRINTTAB(4)"4[SPC]F7[SPC]DIFICIL"
130 PRINT"3.[SPC]PULSA[SPC]F6[SPC]O
  ESPUES[2SPC]DE[SPC]CADA[SPC]PROBLEMA
  [SPC]PARA[SPC]CAMBIAR[SPC]DE[SPC]OP
  ERACION"
140 PRINT"4.[SPC]PULSA[SPC]F4[SPC]PA
  RA[SPC]PONER[SPC]A[SPC]CERO[SPC]EL[SPC]
  MARCADOR"
150 PRINT"5.[SPC]LA[SPC]BARRA[SPC]DE
  [SPC]ESPACIOS[SPC]PARA[SPC]COMENZAR"

200 IF F%>0 THEN GOSUB 2000
210 IF N%>0 THEN GOSUB 3000
220 GETA$: IF A$="" THEN 220
230 IF A$="S" THEN F%=1: YES=0: NO=0: GOSUB
  2000: GOT0220
240 IF A$="R" THEN F%=2: YES=0: NO=0: GOSUB
  2000: GOT0220
250 IF A$="M" THEN F%=3: YES=0: NO=0: GOSUB
  2000: GOT0220
```

```
260 IF A$="D" THEN F%=4: YES=0: NO=0: GOSUB
  2000: GOT0220
270 IF A$=CHR$(133) THEN N%=1: YES=0: NO=
  0: GOSUB 3000: GOT0220
280 IF A$=CHR$(134) THEN N%=2: YES=0: NO=
  0: GOSUB 3000: GOT0220
290 IF A$=CHR$(135) THEN N%=3: YES=0: NO=
  0: GOSUB 3000: GOT0220
300 IF A$=CHR$(136) THEN N%=4: YES=0: NO=
  0: GOSUB 3000: GOT0220
310 IF A$=CHR$(139) THEN 20
320 IF A$=CHR$(138) THEN YES=0: NO=0: GOT
  0220
330 IF A$=CHR$(32) THEN 350
340 GOT0220
350 IF F%=0 OR N%=0 THEN 220
360 FOR I=1 TO 5: RN(I)=INT(RND(X)*10*10%
  ): IF F%=4 THEN RN(I)=RN(I)+1
370 NEXT
380 ON F% GOSUB 500, 700, 900, 1100
390 GOSUB 4000
400 GOSUB 5000
410 PRINT"¡HOY![15CSRSD]"
420 PRINT"PULSA[SPC]ESPACIO[SPC]PARA
  [SPC]OTRO[SPC]PROBLEMA"
430 GETA$: IF A$="" THEN 430
440 IF A$=CHR$(32) THEN 130
450 IF A$=CHR$(139) THEN 20
460 IF A$=CHR$(138) THEN YES=0: NO=0
470 GOT0430
500 RES=0: REM SUMAR
510 PRINTCHR$(147):PRINT"SUMAR".PRIN
  T"-----"
520 FOR I=1 TO N%
530 RES=RES+RN(I)
540 PRINTSPC(6)RIGHT$( "[6SPC]" +STR$(
  RN(I)),N%+2)
550 NEXT
560 RES=RES+RN(N%+1)
570 PRINTSPC(6)RIGHT$( "[4SPC]" +STR$(
  RN(N%+1)),N%+2)
580 PRINTSPC(8)RIGHT$( "-----",N%)
590 RETURN
700 IF RN(1)>RN(2) THEN 720: REM RESTAR
```

¡Hazte socio del Club de Soft de España!  
Tendrás multitud de ventajas...

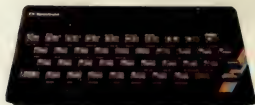
- Recibirás puntual información de las novedades de soft que haya en el mercado.  
Te enviaremos un catálogo bimensual que incluye más de 200 programas para Commodore 64 y Spectrum.
- Podrás adquirir programás con un 30% de descuento, por ser miembro del CLUB DE SOFT DE ESPAÑA.
- Obtendrás descuentos de hasta un 20% sobre hardware.

Si ya eres socio del Club y deseas acogerte a este regalo, puedes hacerlo contra reembolso de 1.000 ptas., pero no olvides decirnos tu número de socio.



## DIFUSION POR AMISTAD

Habla a tus amigos de las enormes ventajas que supone pertenecer al CLUB DE SOFT DE ESPAÑA. Si junto con tu inscripción se inscriben dos amigos tuyos por tu recomendación, te regalamos un programa de juegos.



## ¿COMO PUEDES HACERTE SOCIO DEL CLUB DE SOFT DE ESPAÑA?

- Sólo tendrás que abonar 1.000 Ptas. como cuota UNICA de inscripción.
- Como regalo de bienvenida a nuestro CLUB de SOFT de ESPAÑA podrás elegir entre dos programas:

AJEDREZ	STARSHIP ENTERPRISE
SOLO FLIGHT	SLIPPERY SID
para CBM-64	para SPECTRUM

Te enviaremos el programa elegido completamente GRATIS, junto con la confirmación de tu pertenencia al CLUB de SOFT de ESPAÑA.



## CUPON DE INSCRIPCION AL CLUB DE SOFT DE ESPAÑA

Eraso, 12 - Tel. 24610 94 - 28028-MADRID

Nombre \_\_\_\_\_

Dirección \_\_\_\_\_

Ciudad \_\_\_\_\_ Tel. \_\_\_\_\_

Deseo pertenecer al CLUB DE SOFT DE ESPAÑA. Ruego me envíen como regalo de bienvenida al CLUB el programa \_\_\_\_\_ Firma \_\_\_\_\_

Nº de socio \_\_\_\_\_

El CLUB DE SOFT DE ESPAÑA te enviará el programa solicitado, junto con un reembolso de 1.000 Ptas. más gastos de envío, como cuota de inscripción al Club.

## DIFUSION POR AMISTAD

NOMBRE de un amigo \_\_\_\_\_

Dirección \_\_\_\_\_

Programa de regalo de bienvenida \_\_\_\_\_ Firma \_\_\_\_\_

DIFUSION POR AMISTAD

NOMBRE de un amigo \_\_\_\_\_

Dirección \_\_\_\_\_

Programa de regalo de bienvenida \_\_\_\_\_ Firma \_\_\_\_\_

```

710 B=RN(1)-RN(1)=RN(2)-RN(2)=B
720 PRINTCHR$(147):PRINT"RESTAR" PRI
NT"-----"
730 RES=RN(1)-RN(2)
740 PRINTSPC(6)RIGHT$( "[6SPC]" +STR$(
RN(1))-N%+2)
750 PRINTSPC(6)RIGHT$( "[5SPC]" +STR$(
RN(2))-N%+2)
770 PRINTSPC(8)RIGHT$( "-----",N%)
780 RETURN
900 PRINTCHR$(147):PRINT"MULTIPLICAR
" PRINT"-----":REM MULTIPLICA
R
910 RES=RN(1)*RN(2)
920 PRINTSPC(6)RIGHT$( "[6SPC]" +STR$(
RN(1))-N%+2)
930 PRINTSPC(6)RIGHT$( "[5SPC]" +STR$(
RN(2))-N%+2)
950 PRINTSPC(8)RIGHT$( "-----",N%)
960 RETURN
1100 PRINTCHR$(147):PRINT"DIVIDIR" P
RINT"-----":REM DIVIDIR
1110 IFRN(1)/RN(2) THEN1130
1120 B=RN(1)/RN(1)=RN(2)/RN(2)=B
1130 RES=INT(RN(1)/RN(2))
1140 RC=RN(1)-RES*RN(2)
1150 PRINTSPC(6)+N%,"[COMM][CRSRD]"
,RIGHT$( "[7COMM][CRSRD]",N%+3)
1160 PRINTSPC(5)RIGHT$( "[6SPC]" +STR$(
RN(1))-N%+1)"[CRSRD]"RIGHT$(STR$(RN
(2))-N%+1)
1170 RETURN
2000 PRINT"[HOM][4CRSRD]".
2010 FORK=1TO4 PRINTTAB(3)"[SPC]" NE
XT
2020 PRINT"[HOM][4CRSRD]",TAB(3+40*(
F%-1))"*"
2030 RETURN
3000 PRINT"[HOM][9CRSRD]".
3010 FORI=1TO4 PRINTTAB(3)"[SPC]" NE
XT
3020 PRINT"[HOM][9CRSRD]".
3030 PRINTTAB(3+40*(N%-1))"*"
3040 RETURN
4000 IFX<4THENINPUT ANS:RC=0:RI=0:G
OTO4020:REM RESPUESTA
4010 PRINT INPUT"COCIENTE",ANS INPUT
"RESTO",RI
4020 IF ANS=RES AND RC=RI THEN YES=Y
ES+1: M=1:GOTO4040
4030 NO=N0+1: M=0
4040 PRINT PRINT"[7CRSRD]BIEN=",YES,
"MM="NO
4050 RETURN
5000 PRINT"[HOM][9CRSRD]".
5010 IFM=1THENPRINT"CORRECTO!!" PRIN
T:GOTO5060
5020 IFX<4THENPRINT"LO[SPC]SIENTO[SPC]
PERO[SPC]JERA "RES:PRINT:GOTO5060
5030 PRINT"LO[SPC]SIENTO, [SPC]LA[SPC]
PERO[SPC]LA[SPC]SOLUCION[SPC]JERA "
5040 PRINTSPC(3)+COCIENTE "RES
5050 PRINTSPC(3)+RESTO "RC:PRINT
5060 IFYES=10ANDNO=0ANDN%<4THEN5200
5070 IFYES=20ANDNO=0ANDN%<4THEN5210
5080 IFYES=10ANDNO=2ANDN%<5THEN5220
5090 IFYES=10ANDNO=5ANDN%<10THEN523
0
5100 IFNO=10ANDYES<NOANDN%>1THEN5240
5110 IFNO=10ANDYES<NOANDN%=1THEN5250
5120 RETURN
5200 PRINT"LO[SPC]JESTAS[SPC]HACIENDO
[SPC]BIEN, POR[SPC]QUE[SPC]NO[SPC]JAU
MENTAS[SPC]LA[SPC]DIFICULTAD[SPC]O[SPC]
"
```

```

5201 PRINT"CAMBIAS[SPC]DE[SPC]OPERAC
ION"[SPC]F%":RETURN
5210 PRINT"ERES[SPC]DEMA, [ADOS[SPC]BU
ENO[SPC]PARA[SPC]MI, PRUEBA[SPC]CON[SPC]
OTRA[SPC]OPERACION[SPC]O[SPC]DE[SPC]
QUE[SPC]".
5211 PRINT"LO[SPC]HAGA[SPC]OTRO" RET
URN
5220 PRINT"LO[SPC]SIENTAS[SPC]HACIENDO
[SPC]BIEN, PERO[SPC]TIENES[6SPC]ALGUN
OS[SPC]".
5221 PRINT"FALLOS, [SPC]SUFIERZTE[SPC]
MAS, " :RETURN
5230 PRINT"NO[SPC]LO[SPC]HACES[SPC]MI
AL, [SPC]PERO[SPC]JESTAS[SPC]COMETIENDO
O[SPC]DEMAS[SPC]RADOS[SPC]ERRORES" RETUR
N
5240 PRINT"NO[SPC]CREO[SPC]QUE[SPC]JES
TES[SPC]PREPARADO[SPC]PARA[SPC]JESTE
[SPC]NIVEL, POR[SPC]FAVOR, ELIJE[SPC]
".
5241 PRINT"UN[SPC]NIVEL[SPC]MAS[SPC]
BAJO, " :RETURN
5250 PRINT"POR[SPC]FAVOR, [SPC]PREGUN
TA[SPC]JAC[SPC]TUS[SPC]PREGRES[SPC]O[SPC]
AL[SPC]TU[SPC]PROFESOR." RETURN
```

#### VIC-20 SIN AMPLIAR

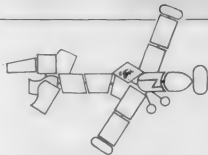
```

5 REM MATEMATICAS VIC-20
10 YES=0: NO=0: F%=0: N%=0
20 PRINTCHR$(147):"[25PC]MATEMATICAS[25PC]
":PRINT PRINT"REM OPCIONES
30 PRINT"1. [SPC]ELIJE[SPC]UNA[SPC]OP
ERACION[PULSANDO
40 PRINTTAB(4)"[6SPC]=[SPC]SUMAR"
50 PRINTTAB(4)"[R[SPC]=[SPC]RESTAR"
60 PRINTTAB(4)"[H[SPC]=[SPC]MULTIPLIC
AR"
70 PRINTTAB(4)"[D[SPC]=[SPC]DIVIDIR"
80 PRINT"2. [SPC]ELIJE[SPC]UNIVEL[SPC]
DE[SPC]DIFICULTAD[SPC]PULSANDO"
90 PRINTTAB(4)"F1[SPC]=[SPC]FACIL"
100 PRINTTAB(4)"F3[SPC]=[SPC]NO[SPC]
TAN[SPC]FACIL"
110 PRINTTAB(4)"F5[SPC]=[SPC]DIFICIL
"
120 PRINTTAB(4)"F7[SPC]=[SPC]MUY[SPC]
DIFICIL"
130 PRINT"3. [SPC]PULSA[SPC]F6[SPC]O
ESPUESTA[SPC]DE[SPC]CHADA[SPC]PROBLEMA
PARA[SPC]CAMBIAR[SPC]DE[SPC]OPERACI
ON"
140 PRINT"4. [SPC]PULSA[SPC]F4[SPC]PA
RA[SPC]PONERLASPCEROT[SPC]EL[SPC]MA
RCADOR"
150 PRINT"5. [SPC]LA[SPC]BARRA[SPC]DE
[SPC]ESCAPTOPARA[SPC]COMENZAR"
200 IFX<0THENGOSUB2000
210 IFN%>0THENGOSUB3000
220 GET$ IFA$="" THEN220
230 IFA$="S" THENF%+1 YES=0: NO=0: GOSU
B2000:GOTO220
240 IFA$="R" THENF%+2 YES=0: NO=0: GOSU
B2000:GOTO220
250 IFA$="H" THENF%+3 YES=0: NO=0: GOSU
B2000:GOTO220
260 IFA$="D" THENF%+4 YES=0: NO=0: GOSU
B2000:GOTO220
270 IFA$=CHR$(133) THENN%+1: YES=0: NO=
0: GOSUB3000:GOTO220
280 IFA$=CHR$(134) THENN%+2: YES=0: NO=
0: GOSUB3000:GOTO220
```

```

290 IF#=#CHR#(135) THEN#N#=#3: YES=#0: NO=#
0: GOSUB3000: GOT0220
300 IF#=#CHR#(136) THEN#N#=#4: YES=#0: NO=#
0: GOSUB3000: GOT0220
310 IF#=#CHR#(139) THEN#20
320 IF#=#CHR#(136) THEN#YES=#0: NO=#0: GOT
0220
330 IF#=#CHR#(32) THEN#350
340 GOT0220
350 IFF#=#0: #N#=#0 THEN#220
360 FORI=1T05: RN(I)=INT(RND(X)*10*#N#)
: IFF#=#4 THEN#RN(I)=RN(I)+1
370 NEXT
380 ONF#GOSUB500,700,900,1100
390 GOSUB4000
400 GOSUB5000
410 PRINTCHR#(19)TAB(220)TAB(220)
420 PRINT"PULSASPCJESPAID(SPC)PARA
[4SPC]OTRO[SPC]PROBLEMA"
430 GET# : IF#=# THEN#430
440 IF#=#CHR#(32) THEN#360
450 IF#=#CHR#(139) THEN#20
460 IF#=#CHR#(138) THEN#YES=#0: NO=#0
470 GOT0430
500 RES=#0: REM SUMAR
510 PRINTCHR#(147): PRINT"SUMAR": PRIN
T"-----"
520 FORI=1T0#N#
530 RES=RES+RN(I)
540 PRINTSPC(6)/RIGHT#("[6SPC]" +STR#(
RN(I)),#N#+2)
550 NEXT
560 RES=RES+RN(#N#+1)
570 PRINTSPC(6)/RIGHT#("[4SPC]" +STR#(
RN(#N#+1)),#N#+2)
580 PRINTSPC(8)/RIGHT#("-----",#N#)
590 RETURN
700 IFRN(1)>RN(2) THEN#720: REM RESTAR
710 B=RN(1): RN(1)=RN(2): RN(2)=B
720 PRINTCHR#(147): PRINT"RESTAR": PRI
NT"-----"
730 RES=RN(1)+RN(2)
740 PRINTSPC(6)/RIGHT#("[6SPC]" +STR#(
RN(1)),#N#+2)
750 PRINTSPC(6)/RIGHT#("[5SPC]" +STR#(
RN(2)),#N#+2)
770 PRINTSPC(8)/RIGHT#("-----",#N#)
780 RETURN
900 PRINTCHR#(147): PRINT"MULTIPLICAR
": PRINT"-----": REM MULTIPLICA
R
910 RES=RN(1)*RN(2)
920 PRINTSPC(6)/RIGHT#("[6SPC]" +STR#(
RN(1)),#N#+2)
930 PRINTSPC(6)/RIGHT#("[5SPC]" +STR#(
RN(2)),#N#+2)
950 PRINTSPC(8)/RIGHT#("-----",#N#)
960 RETURN
1100 PRINTCHR#(147): PRINT"DIVIDIR": P
RINT"-----": REM DIVIDIR
1110 IFRN(1)>RN(2) THEN#1130
1120 B=RN(1): RN(1)=RN(2): RN(2)=B
1130 RES=INT(RN(1)/RN(2))
1140 RC=RN(1)-RES*RN(2)
1150 PRINTSPC(6)+#N#, "[COMM]ICRSRD]"
: RIGHT#("[7COMM]12CRSRU",#N#+3)
1160 PRINTSPC(5)/RIGHT#("[6SPC]" +STR#(
RN(1)),#N#+1): "[CRSR]" +RIGHT#(STR#(RN
(2)),#N#+1)
1170 RETURN
2000 PRINTCHR#(19)TAB(110):
2010 FORK=1T04: PRINTTAB(3) "[SPC]": NE
XT
2020 PRINTCHR#(19)TAB(113+22*(#K-1))
: #
2030 RETURN
3000 PRINTCHR#(19)TAB(242)

```



```

3010 FORI=1T04: PRINTTAB(3) "[SPC]": NE
XT
3020 PRINTCHR#(19)TAB(242):
3030 PRINTTAB(3+22*(#N-1)) "*"
3040 RETURN
4000 IFF#=#4 THEN#PRINTSPC(5): INPUT RN
S: RC=#0: RI=#0: GOT04020: REM RESPUESTA
4010 PRINTTAB(22): INPUT"COCIENTE",RN
S: INPUT"RESTO",RI
4020 IF RNS=RES AND RC=RI THEN YES=Y
ES+1: #M=1: GOT04040
4030 NO=#0+1: #M=#0
4040 PRINTCHR#(19)TAB(220)TAB(66): "BI
EN",YES,"MAL",NO
4050 RETURN
5000 PRINTCHR#(19)TAB(220)
5010 IF#=#1 THEN#PRINT"CORRECTO!!": PRIN
TAB(66): GOT05060
5020 IFF#=#4 THEN#PRINT"LO[SPC]SIENTO[SPC]
PERO[SPC]JERA": PRINTSPC(6)/RES: PRIN
T"AB(44): GOT05060
5030 PRINT"LO[SPC]SIENTO, [SPC]LA[SPC]
PERO[SPC]JERA"
5040 PRINTSPC(3)"COCIENTE",RES
5050 PRINTSPC(3)"RESTO",RC: PRINTTAB
(22)
5060 IFYES=10ANDNO=0AND#N#<4 THEN#5200
5070 IFYES=20ANDNO=0AND#N#<4 THEN#5210
5080 IFYES=10ANDNO>5ANDNO<5 THEN#5220
5090 IFYES=10ANDNO>5ANDNO<10 THEN#523
0
5100 IFNO=10ANDYES<NOAND#N#<1 THEN#5240
5110 IFNO=10ANDYES<NOAND#N#<1 THEN#5250
5120 RETURN
5200 PRINT"LO[SPC]ESTAS[SPC]HACIENDO
[SPC]BIENPOR[SPC]QUE[SPC]NO[SPC]HUME
NTAS[SPC]LA[SPC]DIFICULTAD[SPC]OTRO[SPC]
CAMBIAS"
5201 PRINT"DE[SPC]OPERACION? [2SPC] (F
6)": RETURN
5210 PRINT"ERES[SPC]DEMASIADO[SPC]BU
ENO[2SPC]PARA[SPC]MI. PRUEBA[SPC]CON[4SPC]
OTRA[SPC]OPERACION[SPC]OTRO[SPC]DEJA[SPC]
QUE[SPC]":
5211 PRINT"LO[SPC]HAGA[SPC]OTRO": RET
URN
5220 PRINT"LO[SPC]ESTAS[SPC]HACIENDO
[SPC]BIEN, PERO[SPC]TIENES[SPC]ALGUNO
S"
5221 PRINT"FALLOS, [SPC]ESFUERZATE[SPC]
MAS": RETURN
5230 PRINT"NO[SPC]LO[SPC]HACES[SPC]M
AL, [SPC]PERO[SPC]ESTAS[SPC]COMIEN
DO[6SPC]DEMASIADOS[SPC]ERRORES": RETUR
N
5240 PRINT"NO[SPC]CREO[SPC]QUE[SPC]E
STES[5SPC]PREPARADO[SPC]PARA[SPC]EST
E[5SPC]NIVEL. PUEDE[SPC]FACILMENTE, ELIJE"
5241 PRINT"UN[SPC]NIVEL[SPC]MAS[SPC]
BAJO(F6)": RETURN
5250 PRINT"POR[SPC]FACILMENTE, [SPC]PREGUN
TA[SPC]ALSPC]TUS[SPC]PADRES[SPC]O[SPC]
ALSPC]TUS[SPC]PROFESOR.": RETURN

```

# MusiCalc™

*El Sistema Creativo de Música  
(para personas aún sin conocimientos musicales)*

## ¡Que suene la música!

Convierta a su Commodore 64 en un sofisticado instrumento musical, y él le convertirá en un compositor, director y músico.

Con **MUSICALC**, el sistema de música creativa, toda la familia es capaz de componer y tocar música de una manera instantánea.

¡**MUSICALC** le divierte jugando con la música!

Obtenga copias en papel de sus propias partituras, utilice más de 30 escalas distintas, conéctelo a su equipo de alta fidelidad y a una caja de ritmos, y muchas, muchas posibilidades más.

# SUPER SKETCH™

PERSONAL PERIPHERALS, INC.

**TABLA GRÁFICA**  
para Commodore 64



**Entra en el apasionante mundo del arte de ordenador.  
¡Fácil de usar!**

Se entrega junto con soft en cartucho. Gráficos en alta resolución. 16 colores. 2 páginas de dibujo.

Zoom, menús de diseño, archivo en cinta o disco, posibilidad de incluir texto. Permite printar los gráficos mediante soft adicional.

# SUPER STIK™

PERSONAL PERIPHERALS, INC.

**Mando de juego**



## Durabilidad y buen precio

Compatible con C-64, Atari, Spectravideo, Sinclair Spectrum (con interface) etc...

## MONITOR DE COLOR 14" Profesional



- Pantalla de gran resolución
- Sonido incorporado
- Entrada señal video compuesta o RGB
- Perna orientable
- Gran resistencia y durabilidad

Adaptable a todos los ordenadores.

(Los microordenadores que sólo tienen salida RF son compatibles mediante una sencilla conexión)

ESTAREMOS EN JUVENALIA,  
EN MADRID  
Y EN EL SALON DE LA INFANCIA  
Y JUVENTUD DE BARCELONA

Distribuidor exclusivo

**IDEALOGIC™**

Gran Vía Carlos III, 97 K. 08028 BARCELONA  
Tel.: 330 33 06 - 330 33 62. Télex: 50156 JONG



## Le presentamos a SPINNAKER. Para que también nuestros hijos aprendan jugando con el ordenador.

Muchos padres ya conocen que el ordenador doméstico puede servir para que nuestros hijos jueguen y pasen largos ratos de ocio. Con SPINNAKER los **juegos de ordenador** además sirven **para aprender**.

SPINNAKER es la firma líder mundial en juegos educativos para ordenador. Ahora llega a nuestro país para que también nuestros hijos aprendan jugando con el ordenador.

Nuestros juegos están llenos de emoción, entretenimiento, color, música... Pero además poseen un alto grado de valor educativo, ayudando a los niños a que desarrollen sus habilidades y aprendan de una divertida manera. Aprovechando al máximo las nuevas posibilidades que nos ofrece el mundo de las computadoras.

En su elaboración han intervenido especialistas en la educación, la didáctica, la música, los gráficos y la programación, que han rea-

lizado un cuidado producto especialmente pensado para la educación.

Estos son los motivos por los que miles de niños de todo el mundo son verdaderos fans de SPINNAKER. Y sus padres también.

Si Ud. realmente desea que su ordenador familiar sea una ayuda útil para el futuro de sus hijos, recuerde este nombre: SPINNAKER. Una divertida familia de juegos educativos.



DISPONEMOS DE VERSIONES PARA APPLE II, Commodore 64 y Sinclair Spectrum

Precio especial de  
introducción  
**2.400 pts.**

**IDEALOGIC®**  
Gran Via Carlos III, 97 K. 08028 BARCELONA

**SPINNAKER™**  
SPINNAKER SOFT COR. CAMBRIDGE, MASS. USA



Por Jordi SASTRE

## Rutinas Basic (3)

# Paginación

El tipo más frecuente de listado que se realiza con un ordenador es el típico listado lineal, que se compone de una cabecera y un cuerpo. El cuerpo está formado por un número determinado de líneas (siempre las mismas en cada página), en que cada línea corresponde a un registro. Todas las líneas tienen la misma estructura, aunque sus contenidos son diferentes por pertenecer a diferentes registros.

Cuando se programan estos listados hay dos aspectos que siempre son engorrosos: el primero es calcular, formatear y alinear adecuadamente las columnas del mismo. El segundo es implementar un contador de líneas para paginar en el momento preciso (ni una línea antes ni una después).

La rutina de PAGINACION que presento en este artículo cumple las siguientes funciones: cada vez que se llama (GOSUB 27100) incrementa un contador de líneas (CL). Si este contador no llega a 63 no hace nada más. En cambio, cuando se llega a sobrepasar la línea 63 imprime una serie de interlíneas hasta llegar a las 66 que se supone tiene la hoja de listado, incrementa el contador de páginas (CP) y confecciona la cabecera de listado para continuar listando.

De esta manera, el programador puede olvidarse de la paginación. Sólo debe preocuparse de efectuar el GOSUB después de imprimir cada línea, y la rutina hará todo lo demás.

La rutina de paginación es corta, pero debe manejarse con cuidado, pues maneja bastantes variables:

Y1\$, Z1\$, T\$, y T\$ son cuatro variables que pueden asignarse una vez durante el programa (por ejemplo al principio del mismo) y después ya no preocuparse de ellas.

Y1\$ debe contener el carácter de

expansión de caracteres de la impresora (CHR\$(1) en impresoras Commodore). Si la impresora no posee modo expandido, o no interesa hacerlo, puede quedar como cadena nula (Y1\$="").

Z1\$ debe contener 80 guiones que se utilizarán para trazar las líneas horizontales (si se desea puede utilizarse otro carácter).

T\$ puede contener (no pasa nada si es una cadena nula) el nombre de la empresa que utiliza la programación, o el título del programa, o cualquier texto común a toda la programación.

T\$ puede contener la fecha del día (unos 8 caracteres, por ejemplo: DD. MM. AA.).

Estas variables son las comunes a todos los listados que confeccione un programa. Las variables particulares de cada listado, que debe asignarse antes de la primera llamada a la rutina, son las siguientes:

T0\$ debe contener el título del listado (máximo 40 caracteres).

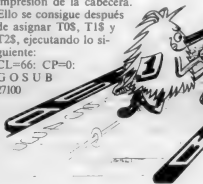
T1\$ es la primera línea de cabecera.

T2\$ es la segunda línea de cabecera (si sólo hay una T2\$="").

También antes de la primera llamada a la rutina deben inicializarse los contadores y forzar una primera impresión de la cabecera.

Ello se consigue después de asignar T0\$, T1\$ y T2\$, ejecutando lo siguiente:

```
CL=66: CP=0:
G O S U B
27100
```



### EJEMPLO:

Hace dos meses, en el primer artículo de esta serie, cuando hablábamos de la formatación numérica, puse un ejemplo de formatación utilizando un programa que imprimía diez líneas con tres columnas. Reproduzco aquí el mismo programa, pero incluyendo cabecera y rutina de paginación.

Es obvio que cuanto más estrecho sea el listado, más cortos deberán ser los contenidos de las variables T\$ y T\$. Para comprobar eficazmente el buen funcionamiento de la rutina, obligando a imprimir varias hojas, basta con modificar la línea 210 del programa, poniendo el bucle FOR...NEXT a 150, por ejemplo (210 FOR N=1 TO 150), para llenar dos páginas y media. Asimismo también conviene cambiar la línea 130 para que en lugar de elevar a la séptima potencia lo haga sólo a la cuarta (N^4), puesto que 150^7=1.70859375E+15, lo cual excede de la precisión de la CPU.

### Explicación de la rutina

Veamos ahora cómo trabaja la rutina, línea a línea:

27105: Cálculo del ancho del listado, definido por la longitud de T\$

27110: Incrementa contador de líneas. Si no llega a 63 vuelve a la rutina.

27120: Imprime las interlíneas necesarias para llegar al principio de la siguiente página.

27130: Incrementa el contador de páginas.

27135 y 27138: Imprime T\$, la fecha (T\$) y el número de página (CP).

27140 y 27145: Imprime el título del listado centrado, con carácter expandido si Y1\$ está asignado o con espacios en blanco intercalados si no lo está.

27150: Imprime la primera línea de cabecera (T1\$) si está asignada.

27160: Imprime la segunda línea de cabecera (T2\$) si está asignada.

27170: Imprime una línea horizontal como fin de cabecera y retorna.

### Observaciones

Esta rutina trabaja de un modo muy peculiar, por lo que su uso sofisticado

puede requerir cierta práctica. No obstante, hay algunos puntos que se pueden comentar de antemano.

Los listados no han de ser forzosamente de 80 ó 132 columnas, sino que su ancho es variable. La rutina toma como ancho la longitud de la cadena T\$.

Si el listado sólo tiene una línea de cabecera, la variable T2\$ debe dejarse como nula: T2\$=""



Si no se desea ningún texto de cabecera, la variable T1\$ debe asignarse con tantos espacios en blanco como ancho deba ser el listado.

Si no se desea que aparezca el número

de página, porque ya se sabe de antemano que el listado sólo tendrá una, puede ponerse CP=1 en lugar de CP=0 en la inicialización del listado (obsérvese líneas 27130 y 27135).

#### Programa "Paginación"

```
10 REM *****
20 REM *
30 REM * PAGINACION *
40 REM *
50 REM *****
60 REM
70
80 REM DATOS DE ENTRADA
90 REM
100 REM
110 REM T0$ = TÍTULO DEL LISTADO
120 REM T1$ = LINEA PRIMERA DE CABEC
ERA
130 REM T2$ = LINEA SEGUNDA DE CABEC
ERA
140 REM
150 REM CONTADORES INTERNOS
160 REM
170 REM
180 REM CL = CONTADOR DE LINEAS
190 REM CP = CONTADOR DE PAGINAS
200 REM
210 REM FUNCION DE LA RUTINA
220 REM
230 REM
240 REM INCREMENTA EL VALOR DE LAS
    LINEAS.SI SE LLEGA A 63
250 REM COMIENZA UNA NUEVA PAGINA
    IMPRIMIENDO LA CABECERA.

260 REM TÍTULO DEL LISTADO, NUMERO
    DE PAGINA,...
270 REM
330 REM OBSERVACIONES
340 REM
350 REM
360 REM DEBEN ESTAR ASIGNADAS LAS
370 REM VARIABLES V1$,Z1$,TE$ Y TF$.

380 REM POR EJEMPLO, ANTES DEL PROGR
    AMA
    DEBE EJECUTARSE:
390 REM 10 Z1$="" :FORX=1TO80:Z1$=Z
    1$+"-":NEXT
400 REM 20 V1$=CHR$(1)
410 REM 30 TE$="NOMBRE(SPC)DE(SPC)
    EMPRESA(SPC)PROGRAMACION"
420 REM 40 TF$="26.11.84":REM FECH
    A
430 REM
440 REM ** PAGINACION *****
450 REM
460 REM V=LEN(T1$)+1
470 REM CL=CL+1 IFCL<63THENRETURN
4710 IFCL<63THENFORX=CL+1TO66:PRINT
    #4:NEXT
4710 CP=CP+1
4715 PRINT#4:PRINT#4,TE$(SPC(Y-20-LE
    N(T$)))TF$,IFCP THENPRINT#4,"(2SPC)P
    AGINA"CP,
4718 PRINT#4:PRINT#4
47140 X1=INT((Y-LEN(T0$)+2)/2):IFY1$
    <" THENPRINT#4,SPC(X1)+V1$TO6:GOTO2
    7150
47145 PRINT#4,SPC(X1),FORX=1TOLEN(T
    0$):PRINT#4,MID$(T0$,X,1)$(SPC)",NE
    XT:PRINT#4
47150 PRINT#4,CL=6:IFT1$<" THENPRIN
    T#4,T1$:CL=CL+1
```

```
27160 IFT2$<" THENPRINT#4,T2$:CL=CL
+1
27170 PRINT#4,LEFT$(Z1$+Z1$,Y-1):RET
URN
```

#### Ejemplo de paginación + formatación

```
1 T0$="EJEMPLO(SPC)DE(SPC)PAGINACION
[SPC]+[SPC]FORMACION"
2 T1$="(72SPC)"
3 T2$=""
4 V1$=CHR$(1)
5 TE$="CLUB(SPC)COMMODORE"
6 TF$="26.11.84"
10 Z2$="" :FORX=1TO80:Z2$=Z2$+"[SPC]"
:NEXT
15 Z1$="" :FORX=1TO80:Z1$=Z1$+"-":NEXT
20 OPEN#4:REM LISTADO POR IMPRESORA

30 CL=66:CP=0:GOSUB27100:REM FUERZA
PAGINACION
100 FORN=1TO150
110 B=N:MX=5:GOSUB26299
120 B=N*7:MX=12:GOSUB26299
130 B=LOG(N):MX=10.5:GOSUB26299
140 GOSUB27100:PRINT#4:NEXT
150 END
26200 REM * FORMAT.MILES *****
***
26205 MY=INT(MX):MD=INT(MX*10-MY*10)
:IFMD=0THEN26270
26210 X=3:X1=INT(10*MD+.5):B=SGN(B)*
INT(ABS(B)*X1+.5)/X1:X$="" :IFB<0THEN
X$="-"
26215 B2=ABS(B):B1=INT((B2-INT(B2))*
X1+.5):B$=STR$(INT(B2))
26220 B$=RIGHT$(B$,LEN(B$)-1)
26225 IFLEN(X$)+LEN(B$)+INT(LEN(B$)
-1)/3)+MD+1<MYTHEN26235
26230 B$="" :FORX=1TOY:B$=B$+"$":NEXT
T:RETURN
26235 IFLEN(B$)>XTHENB$=LEFT$(B$,LEN
(B$)-X)+".":RIGHT$(B$,X):X=X+4:GOTO2
6235
26240 IFB1=0THENB1$="" :LEFT$(Z1$,MD)
:GOTO26260
26245 B1$=STR$(B1):B1$="" :RIGHT$(B1$)
0000000000+RIGHT$(B1$,LEN(B1$)-1),MD)
26255 B1$=LEFT$(B1$+"0000000000",MD+1)
)
26260 B$=RIGHT$(Z2$+X$+B$,MY-MD-1)+B
1$:RETURN
26270 B=SGN(B)*INT(ABS(B)+.5):B$=STR
$(ABS(B)):B$=RIGHT$(B$,LEN(B$)-1)
26275 IFLEN(B$)>3THENB$=LEFT$(B$,LEN
(B$)-3)+".":RIGHT$(B$,3)
26280 IFLEN(B$)>7THENB$=LEFT$(B$,LEN
(B$)-7)+".":RIGHT$(B$,7)
26285 IFB<0THENB$="-"+B$
26290 IFLEN(B$)>MYTHENB$="" :FORX=1TO
MY:B$=B$+"$":NEXT:RETURN
26295 B$=RIGHT$(Z2$+B$,MY):RETURN
26299 GOSUB26200:PRINT#4,B$:RETURN
27100 REM ** PAGINACION *****
***
```

## MONITOR TM 01

Monitor de 14 pulgadas de rápido calentamiento con 90° de deflexión, asento de corriente estática, tubo pantalla-vergente de color.

Señales de entrada seleccionables mediante interruptor:

- RGB lineal a 1 voltio p-p
- YUV lineal
- RGB TTL

Transformable en pantalla verde. Alta resolución (585 × 436).

**P.V.P. 59.500 ptas.**



## MONITOR TM 80

Monitor de 12 pulgadas fósforo verde P-31, antirreflexivo.

Alta resolución, 900 líneas en centro de pantalla.

**P.V.P. 25.000 Ptas.**



SE  
BUSCAN  
DISTRIBUIDORES

Ideales para ser utilizados por una gran gama de micro-ordenadores tanto domésticos como profesionales.

Al accionar el interruptor de encendido los monitores quedan preparados instantáneamente para la recepción de datos.

Estos monitores pueden usarse con: Commodore, Atari 800, modelos de la BBC, Dragón, Lynx, Oric, Apple II y IIe (con columna RGB de color), Positrón 9000 y Sinclair QL.

Diseñado y producido por TATUNG. UK (REINO UNIDO)

Importador exclusivo para España.

Gran Vía Carlos III, 86-6º

08028 Barcelona

**ALPHA MUNDIAL GROUP, S.A.**

Teléfono 330 96 51  
Telex 52220 ARTRA



```

27105 Y=LEN(T1$)+1
27110 CL=CL+1:IFCL<63THENRETURN
27120 IFCL<67THENFORX=CL+1TO66:PRINT
#4:NEXT
27130 CP=CP+1
27135 PRINT#4,PRINT#4,TE$SPC(Y-20-LE
NKTE$>)*FS$:IFCPTHENPRINT#4,"[25PC]P
ROINA"CP:
27138 PRINT#4:PRINT#4
27140 X1=INT((Y-LEN(T0$)*2)/2):IFY1$
<>"THENPRINT#4,SPC(X1)Y1$T0$:GOTO27
150
27145 PRINT#4,SPC(X1):FORX=1TOLEN(T

```

```

0$):PRINT#4,MID$(T0$,X,1)"[SPC]",NE
XT:PRINT#4
27150 PRINT#4:CL=6:IFT1$<>"THENPRIN
T#4,T1$ CL=CL+1
27160 IFT2$<>"THENPRINT#4,T2$ CL=CL
+1
27170 PRINT#4,LEFT$(ZI$+ZI$,Y-1):RET
URN

```

CLUB COMMODORE

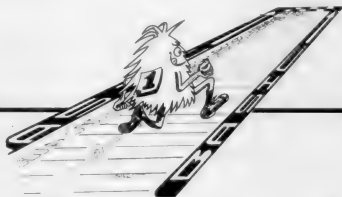
26.11.84 PAGINA 1

EJEMPLO DE PAGINACION + FORMATAACION

```

1      1      0 -----
2      128    0 69915
3      2.187  1 03961
4      16.384 1' 38629
5      78.125 1' 60944
6      272.356 1' 79176
7      823.543 1' 94591
8      2.097.152 2' 07944
9      4.732.359 2' 13722
10     10.000.000 2' 30259
11     19.487.171 2' 39730
12     35.831.808 2' 43431
13     62.748.517 2' 56495
14     105.413.504 2' 63906
15     170.859.375 2' 70805
16     268.435.456 2' 77259
17     410.398.674 2' 83321
18     612.220.833 2' 89037
19     893.871.739 2' 94444

```



## ELECTROAFICION COMPUTER

C/ VILLARROEL, 104 - 08011 BARCELONA - TEL: 253 76 00 - 09

C/ GRAN VIA CORTS CATALANES, 559 - 08011 BARCELONA - TEL: 254 23 19

### PRODUCTOS COMMODORE

Commodore 64  
Disk Drive 1541  
Cassette C2  
Monitor Color 1701  
Impresora MPS-801  
Commodore 64SK Portable  
VIC-20  
Commodore 16

### SOFTWARE

Contabilidad  
Contabilidad Doméstica  
Control de Stocks  
Mailing y Etiquetas  
Ficheros  
Base de Datos  
Gran variedad de Juegos  
Programas Educativos

### GAMA COMPLETA DE ACCESORIOS

Interfaces  
Joysticks  
Sintetizadores de voz  
Cassettes  
Cintav  
Discos  
Base de Datos  
Easy Script  
Monitores  
Interpod  
Cables  
Procesador de Textos  
Libros

### SINCLAIR

Spectrum 48K  
Impresora Seikosha  
con interface  
Microdrive  
Teclado DR'TRONICS  
LAPIZ óptico  
Amplificador Sonido

### IMPRESORAS

Seikosha  
Star  
Epson  
NewPrint  
C. Irah  
Riteman

### ORDENADORES DE GESTION

Amstrad  
Pal Computer  
Commodore  
Apple

Por Pere MASATS

# Una serie de utilidades y trucos

*Es muy frecuente —incluso entre programadores profesionales— cometer errores al entrar un programa, especialmente cuando se trata de sentencias data. A continuación damos algunas ideas de cómo encontrar estos errores.*



cadena que se dan como sentencias DATA para beneficiarse de las facilidades que proporciona el editor de pantalla de los equipos COM-MODORE. Lo

que es una lata en la utilización de DATAS en un programa es el hecho de que deben leerse en forma secuencial sin poder retroceder a DATAS anteriores a menos que se utilice RESTORE para volver al principio. RUN, CLR y RESTORE sitúan el puntero de lectura de DATAS (DATA read pointer, 62/63 en

los CBM, 65/66 en VIC-20 y C-64) apuntando al inicio de BASIC. Cuando se efectúa un comando READ el puntero va explorando el texto del programa en busca de una sentencia DATA, si no la encuentra cuando ha llegado al fin del programa se genera un error

## OUT OF DATA

Por otra parte existe otro puntero (direcciones 119/120 en los equipos CBM y 122/123 en el VIC-20/C-64) que le indica al intérprete dónde está el código que se está interpretando en este momento. Si el valor de este puntero se transfiere al puntero de lectura de DATAS se podrá forzar la lectura de DATAS de forma no secuencial. Por ejemplo:

```
10 DATA PRIMERO, SEGUNDO,
    TERCERO
20 DATA CUARTO
30 READ AS, BS
40 POKE65, PEEK(122): POKE66,
    PEEK(123)
50 READ A, B
60 PRINTA$,B$,A,B
70 DATA 1,2,3,4
80 END
```

El comando READ de la línea 30 lee en AS y BS los DATAS PRIMERO y SEGUNDO dejando el puntero en TERCERO. La línea 40 mueve el puntero a través de las líneas 10, 20 y 30 a un lugar de la misma línea 40, como en esta línea no hay una sentencia DATA el próximo READ se efectuará (línea 50) sobre la línea 70.

Las sentencias DATA son muy prácticas cuando se trata de almacenar rutinas en código máquina en la memoria del ordenador, para ejecutarlas luego mediante SYS y trabajar a velocidades razonables en partes críticas del programa. Pero es muy fácil cometer un error en algo tan abstracto como un número y la mayor parte de los principiantes odian a muerte esta manera de introducir datos. Y muchos autores cargan con el sambenito de que sus programas —listados en una revista— no funcionan. Vamos a ver cómo podemos solucionarlo.

Entre en su ordenador el programa siguiente:

```
10 SC=1024: REM PARA EL VIC
    UTILICE 7680 SIN EXPANSION
    O 4096 CON EXPANSION.
20 PRINT"[CLR/HOME]"*****
30 FOR J = SC TO SC + 5
40 READ K: POKE J,K
50 NEXTJ
60 REM (MAS LINEAS DE PRO-
    GRAMA)
500 DATA 20,5,19,20
510 DATA 999
```

En este ejemplo se utiliza la conocida técnica de hacer un READ seguido de un POKE en un bucle FOR/NEXT para escribir TEST en la pantalla. El caso es que si hacemos un RUN de este programa nos encontramos con un ?ILLEGAL QUANTITY ERROR IN 40, pero en la línea 40 no hay ningún error. Veamos: si hacemos PRINT

J:PRINT K, veremos que la variable K está fuera del valor de 0 a 255 que deben tener los valores que se colocan mediante POKE en posiciones de memoria, luego efectivamente hay un error en la línea 40. Es evidente que el error está realmente en la línea 510 y consiste en el valor 999. Buscar a través de una lista de DATAS el valor incorrecto es —además de un verdadero fastidio— algo sólo relativamente fiable, pues nos hemos encontrado con frecuencia que cometemos el mismo error al revisar el programa que se cometió al entrarlo. Para corregir esto se puede recurrir a la ayuda de otra persona que —con "nuevos ojos"— nos ayude a repasar nuestro programa y el listado. Pero existe una manera muy fácil de encontrar el DATA incorrecto. En las posiciones 63 y 64 (VIC-20 y C-64) se almacena el número de línea del DATA que se ha leído en último lugar, así si después del ?ILLEGAL QUANTITY ERROR hacemos (en modo directo):  
PRINT PEEK(63)+PEEK(64)\*256  
obtendremos 510 que es la línea donde REALMENTE está el error.

## Más sobre datas

De hecho es muy práctico incluir en el programa grandes cantidades de datos que el mismo necesita para su funcionamiento, y muchas veces para mejorar la presentación se recurre a determinadas



# COMPARTIENDO EXPERIENCIAS ENTRE AMIGOS

Esta sección está dedicada a la colaboración de todos nuestros lectores y está dividida en dos partes:

1) Programación:

Programas y similares.

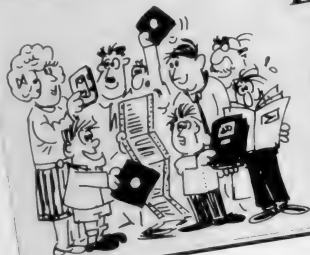
2) Magia:

Trucos, sugerencias, etcétera. (pág. 52).

Habrán premios y alicientes para todos los participantes.

Enviarnos vuestra dirección para que podamos poneros en contacto unos con otros.

¡Animaros, chicos (...y chicas)!



## VIC-20 sin expansión

### Golf

Joaquín Torres

Recordaréis que en el nº 3 de abril apareció un programa para el C-64 titulado GOLF.

Bueno, pues me he decidido y lo he adaptado para el VIC-20. A ver si podéis mejorarlo, sobre todo en el aspecto musical, porque presentación y juego creo que ya valen.

Yo no he podido poner mejor sonido porque no dispongo de la suficiente memoria como para eso, pues sólo me quedan libres de setecientos a ochocientos bytes.

El programa en sí es similar al del C-64. Sólo cambian los POKE y luego las últimas líneas que son la presentación y la espera de respuesta.

Bueno, ya solo deseáis ánimo y a ver si adaptáis más programas. Y por mi parte seguiré intentándolo.

0 REM\*\*\*\*\*

10 REM

20 REM GOLF VIC - 20

30 REM ADAPTACION DE:

40 REM

50 REM JOAQUIN TORRES

60 REM

70 REM 02 - 11 - 84

80 REM

90 REM\*\*\*\*\*

100 POKE36865,155:POKE36879,8:POKE36

878,15:POKE36869,240:PRINTCHR\$(8)

110 PRINT"[CLR]¡WHT!" :GOSUB900

120 V=7680:W=38400

130 FORC=1TO9:Z=176+C

140 PRINT"[CLR]";

150 FORK=0TO21

160 POKEV+(22\*18)+X,69

170 POKEW+(22\*18)+X,1

180 NEXT

190 PRINT"[8SPC]GOLF[18SPC][4COMHT][9SPC]

"

200 PRINT"[EL][SPC]CRMFO[SPC]TIENE[SPC]

20[2SPC]ME-[SPC]TRO5."

210 CL=INT(20\*RND(0)+1):IFCL<1THEN21

0

220 IFCL>21THEN210

230 POKEV+(22\*18)+CL,32

240 POKEV+(22\*18)+CL-21,101

250 POKEV+(22\*18)+CL-43,101

260 POKEV+(22\*18)+CL-65,101

270 POKEV+(22\*18)+CL-87,101

280 POKEV+(22\*18)+CL-109,Z

290 POKEW+(22\*17),81

300 POKEW+(22\*17),1

310 PRINT"[HOM]¡4CRSRDJA[SPC]CUANTOS

[SPC]METROS[SPC]¡ESTAI[SPC]JEL[SPC]HOVO

"C:"

320 INPUT:IFA>20THENPRINT"EXAGERADO

[SPC]¡QUE[SPC]JEL[SPC]CRM-PO[SPC]ISOLO

[SPC]TIENE[2SPC]20[2SPC]ME-TROS." :GO

TO310

330 IFA<0THENPRINT"NO[SPC]SE[SPC]ADM

ITEN[SPC]¡IRADAS[SPC]¡HACIA[SPC]¡DETRA

S." :GOTO310

340 FORH=1TOA

350 POKEV+(22\*17),32

360 POKEV+(22\*17)+H,81

370 POKEW+(22\*17)+H,1

380 POKEV+(22\*17)+H,32

390 NEXTH

400 POKEV+(22\*17)+A,81

410 POKEW+(22\*17)+A,1

420 IFA=CLTHEN440

430 GOTO560

440 POKEV+(22\*19)+A,81

450 POKEW+(22\*19)+A,1

460 POKEV+(22\*17)+A,32

470 FORAZ=7680TO7966:POKEAZ,32:NEXTA

Z

480 PRINT"[HOM]¡2SPC]0.K.[SPC]¡HOVO"C

"HECHO."

490 REM \*\*ACIERTO\*\*

500 FORL=1TO15

510 FORM=220+L\*2TO200STEP-1

520 POKE36876,M

530 NEXTM,L

540 POKE36876,0

550 GOTO720

560 FORAZ=7680TO7966:POKEAZ,32:NEXTA

Z

570 PRINT"[HOM]¡2CRSRD]EL[SPC]¡HOVO[SPC]

ESTABAI[SPC]JA[3SPC]CL"[2SPC]¡METROS."

# SI VAS A SEGUIR VIVIENDO EN ESTE PLANETA NO TE QUEDAS DESCONECTADO

¿Cómo dominar  
a los ordenadores  
antes de que ellos  
dominen todos  
los sectores  
de la vida?

Si dominas  
tu impaciencia,  
ya tienes  
medio curso  
hecho

Nuevo curso a distancia  
Ordenador Personal y  
Programación Basic

Ahora puedes hacerlo de una forma sencilla y divertida. Con un dedo de la mano. Sin saber una palabra de inglés. Y sin moverte de casa.

Gracias a un curso claro y entretenido que te ayuda a sacarle al ordenador todo lo que tiene dentro.

Por supuesto que no te estamos hablando de un Manual de Instrucciones, y allá tú. Se trata de un programa de aprendizaje muy práctico, diseñado por especialistas, para que nadie se aburra a mitad de camino.

Es, para que te hagas una idea, como un viaje organizado al futuro. ¿Quieres una plaza?

Te ofrecemos 12 manuales de lecciones explicadas paso a paso.

Te ofrecemos, si no tienes ordenador, el COMMODORE 64, el más interesante del mundo, para que practiques desde el primer momento. Te ofrecemos un cuadro de profesores que te orientan continuamente en tu aprendizaje. ¿Qué pones tú? Paciencia. Y un poco de voluntad.

Con esos dos instrumentos te garantizamos que en seis meses serás capaz de operar con cualquier programa que adquieres y algo que te gustará más: podrás crear tus propios programas para uso profesional o personal: gestión, administración, archivo, estadística, juegos, estudios... Verás qué divertido es tener paciencia.

Este es el curso diferente a todos los que conoces. Es serio, pero sabiendo que en tres años naufraga. Es divertido pero sin olvidar que tienes que seguirlo partido.

Consta de los siguientes elementos:

- 12 manuales de lecciones, preparados para aprender, enseguida son publicados en francés.
  - 1 año de asesoramiento.
  - Optativo: Un ordenador personal COMMODORE 64, con su unidad de casetes, para practicar desde el primer minuto, porque como se aprende a dominar los ordenadores, es trabajando con ellos.
  - Amplio número de programas y una orientación continuada de los profesores.
  - No son necesarios conocimientos especiales de ningún tipo.
  - Duración: Sesenta días, aproximadamente, dedicando dos horas diarias.
  - Sistema de cuestionaje: A distancia.
- Cada manual incluye explicaciones, ejemplos para realizar en el ordenador. El profesor contesta a vuelta de correo con los cuestionarios y los consejos necesarios.
- Precio: El curso completo que abarca todo del primer al quinto ordenador. Y con facilidades de pago.

CURSOS  
**CCC**

PARA APRENDER  
EN CASA



## Otros de los cursos CCC:

Electrónica  
Radio-TV  
Técnicas en Energía Solar  
Instalación  
Electricidad  
Técnica en Fontanería  
Graduado Escolar  
Guitarra  
Contabilidad  
Mecánica de motos  
- Dibujante de comics.  
- Inglés (con casete o por video)  
Auxiliar Informática  
Puericultura  
Estheticisme  
Peluquería  
Masaje y  
Digitopresión  
Corte y Confección

## Para hablar con los dedos, te echamos una mano

El idioma del futuro, se practica con los dedos. Si quieres aprenderlo, la oportunidad está en tu mano. Envía este cupón.

¡Envía información gratis y sin compromiso  
sobre el curso de:

Nombre y  
apellidos: \_\_\_\_\_  
Dirección: \_\_\_\_\_ N \_\_\_\_\_ Pcia: \_\_\_\_\_  
Publicación: \_\_\_\_\_  
Profesión: \_\_\_\_\_ Cód. Postal: \_\_\_\_\_  
Teléfono: \_\_\_\_\_ Ciudad: \_\_\_\_\_

Envía este cupón a:

CCC, Atto de Manzanillo, Apdo. 666, 20080  
Irujo, (943) 467600 SAN SEBASTIAN  
o a CCC, Apdo. 666, 20080 MADRID

(0-41-41-01-85)



```

580 REM **FALLO**
590 FORL=1T015
600 FORL=230T0200STEP-1:FORM=1T050:N
EXTM
610 POKE36876,L
620 NEXT
630 POKE36877,220:POKE36876,0
640 FORL=15T00STEP-1
650 POKE36878,L
660 FORM=1T0300
670 NEXTM,L
680 POKE36877,0
690 POKE36878,15
700 POKE36876,0
710 PRINT"([CRSRD])(3SPC)MALA(SPC)SUER
TE,([SPC])TRA([2SPC])VEZ([SPC])SERRA.";GOT
0830
720 IFC<9THENNEXTC
730 PRINT"VA(SPC)NO(SPC)HAY(SPC)MAS(SPC)
HOYOS"
740 FORL=1T020
750 FORM=254T0240+INT(RND(1)*10)STEP
-1
760 POKE36876,M
770 NEXTM
780 POKE36876,0
790 FORM=0T0INT:RND(1)*100+120
800 NEXTM,L
810 PRINT"([CRSRD])QUIERES([SPC])OTRA([SPC])
DESDE(SPC)EL([SPC])PRINCIPIO?" ;GOSUB86
0
820 GOT0130
830 FORM=1T02000:NEXT:POKE36879,27:P
RINT"([CLR])(BLU)";CHR$(9);END
840 GOT0130
850 REM *SUBROUTINA DE PREGUNTA
860 GETB$=IFB$=""THEN860
870 IFB$="S"THENRETURN
880 POKE36879,27:PRINT"([CLR)";CHR$(9)
;END
890 REM *SUBROUTINA DE PRESENTACION
900 PRINT"([CLR])(2CRSRD)(4CRSR)JOAQU
INT(SPC)TORRES"
910 PRINT"([HOM])(5CRSRD)(7CRSR)PRESE
NTA"
920 PRINT"([6CRSRD])(RVSON)(SHIFT)(RVSOFF)
(2COMM+)(RVSON)(COMM+)(RVSOFF)(2SPC)

```

```

(RVSON)(SHIFT)(RVSOFF)(2COMM+)(RVSON)
(COMM+)(RVSOFF)(2SPC)(COMM+)(5SPC)(RVSON)
(SHIFT)(RVSOFF)(2COMM+)(RVSON)(COMM+
)(RVSOFF)";
930 PRINT"([COMM+)(2SPC)(COMM+)(2SPC)
(COMM+)(2SPC)(COMM+)(2SPC)(COMM+)(5SPC)
(COMM+)(3SPC)";
940 PRINT"([COMM+)(2SPC)(COMM+)(2SPC)
(COMM+)(2SPC)(COMM+)(2SPC)(COMM+)(5SPC)
(COMM+)(3SPC)";
950 PRINT"([COMM+)(5SPC)(COMM+)(2SPC)
(COMM+)(2SPC)(COMM+)(5SPC)(3COMM+)(SPC)
";
960 PRINT"([COMM+)(SPC)(COMM+)(RVSON)
(COMM+)(RVSOFF)(2SPC)(COMM+)(2SPC)(COMM+
)(2SPC)(COMM+)(5SPC)(COMM+)(3SPC)";
970 PRINT"([COMM+)(2SPC)(COMM+)(2SPC)
(COMM+)(2SPC)(COMM+)(2SPC)(COMM+)(5SPC)
(COMM+)(3SPC)";
980 PRINT"([COMM+)(2SPC)(COMM+)(2SPC)
(COMM+)(2SPC)(COMM+)(2SPC)(COMM+)(5SPC)
(COMM+)(3SPC)";
990 PRINT"([COMM+)(2COMM+)(SHIFT)(2SPC)
(COMM+)(2COMM+)(SHIFT)(2SPC)(COMM+
)(2COMM+)(SHIFT)(2SPC)(COMM+)(3SPC)";
;
1000 REM *SUBE PANTALLA A POSICION N
ORMAL
1010 FORAA=155T038STEP-1
1020 POKE36865,AA
1030 FORBB=0T0125:NEXTBB,AA
1040 FORBB=0T09999:NEXT
1050 PRINT"([CLR)";:RETURN

```

C-64

## Concierto de Aranjuez

José Luis Muñoz Franco  
Monterde, 1-7º B  
50010 ZARAGOZA

Hola a todos los  
lectores de esta mag-  
nífica revista. Aquí  
os mando otro pro-  
grama al que he titu-  
lado EL CONCIERTO DE ARANJUEZ. Se trata de un programa  
totalmente musical que creo que sin ninguna duda os va a gustar a  
todos. El funcionamiento del programa es el siguiente:

En pantalla aparece el título y el teclado completo de un piano  
con 3 octavas y con el número de cada nota de la octava. A continua-  
ción y una vez terminado completamente el dibujo, se dan 2 opciones:  
1: Manual y 2: Automático. Si pulsáis 1, el ordenador hará sonar una  
nota del consabido Concierto de Aranjuez cada vez que vosotros

## GLOSARIO

### Binary Operator.—

Operador bina-  
rio. Un operador arit-  
mético que relaciona dos  
términos. Los operadores binarios  
más utilizados son los de: suma  
(+), resta (—), multiplicación (\*)  
y división (/).

**Binary search.**— Búsqueda bina-  
ria. Es un método de búsqueda  
dicotómica en el que, en cada  
paso, se divide el rango a buscar  
en dos partes iguales y se van  
realizando aproximaciones conse-

cutivas con los  
elementos de los extre-  
mos del intervalo que contiene  
al elemento buscado.

**BISAM.**—Basic indexed se-  
quential access method. Es un  
método de acceso utilizado en  
basic para acceder a datos  
almacenados en unidades que  
permiten acceso directo, loca-  
lizándolos rápidamente por me-  
dio de un índice que generalmen-  
te se guarda en un archivo  
secuencial junto con los datos.  
El índice es actualizado cada vez  
que se hace una modificación  
que afecte a los campos de  
búsqueda por clave (ver artículo

sobre los ficheros ISAM número  
5 de Commodore World).

**BIT.**—Binary digit. En no-  
tación binaria, cualquiera de los  
caracteres 0 ó 1. Es la unidad  
más pequeña posible de infor-  
mación. Un bit es suficiente para  
decir la diferencia entre dos  
elementos opuestos como sí y  
no.

**Bit pattern.**—Configuración de  
bits. Una configuración de N  
dígitos binarios que representan  
una de las 2<sup>N</sup> posibles combi-  
naciones con ese número de bits.  
Por ejemplo con cuatro bits se  
pueden obtener 16 combinacio-  
nes diferentes.







pulséis una tecla y además situará un punto negro sobre la tecla del piano que está sonando en ese momento. De esta forma vosotros sois los que imprimís el ritmo que queráis. Si pulsáis 2, el ordenador hará sonar las notas de esta melodía a su ritmo normal a la vez que también nos indica sobre el teclado del piano la nota que suena en cada momento y suena un acompañamiento (cosa que no ocurre en la opción 1). He de advertiros, aunque ya lo comprobáis, que aunque en pantalla aparece el teclado de un piano, el sonido que se escucha es de guitarra española, así como el de acompañamiento. Y dicho esto paso a explicaros el programa:

#### Lineas:

5-45: Presentación del título y autor (autor del programa, no de la melodía. Qué más quisiera yo). La línea 35 podéis modificarla a vuestro gusto siempre que nombre y apellidos no ocupen más de 13 caracteres ya que si no se saldrían del recuadro.

55-410: Como la sentencia REM indica se trata de dibujar en la pantalla el teclado de un piano. Hubiera sido más fácil usando PRINT, pero queda mucho más bonito hacerlo con POKE ya que el efecto que produce en pantalla da una impresión de sofisticación. La inicialización de la variable numérica A=54272 en la línea 10 resulta muy útil ya que (aunque creo que todos lo sabéis) 54272 es la diferencia entre una posición de la memoria de pantalla y la misma posición de la memoria de color así como la primera posición de memoria de los registros de sonido.

450-555: Ajustes sonoros de las voces 1 y 3 que son las que vamos a utilizar. La 1 para la melodía normal y la 3 para el acompañamiento. Una vez realizados estos ajustes, el sonido obtenido es el de guitarra española.

600: Se encarga de tomar los 90 valores de los 90 datos correspondiente a las 90 notas que componen la melodía. Cada dato esta compuesto de 10 números. Los 3 primeros son la alta frecuencia de cada nota. Los 3 siguientes son la baja frecuencia de cada nota. Los 2 siguientes son la duración de cada nota. Cuando una nota empieza a sonar, el valor de TI es igual a 0. Mientras su valor sea menor que estos 2 números, la nota seguirá sonando. Los 2 números restantes indican el lugar del piano en el que deberá aparecer el punto que indicará qué nota se está tocando en ese momento. A estos 2 números le sumaremos luego 1.864 para que el punto se sitúe en la posición de memoria de pantalla correspondiente con POKE.

610-640: Se encargan de que elijamos una de las 2 opciones anteriormente mencionadas.

700-750: Aquí es cuando empieza a sonar la melodía.

700: Inicializa el bucle que hará sonar las 90 notas y pone el reloj a 0 al principio de cada nota.

704: Calcula el lugar en el que aparecerá el punto sobre el piano.

705: Hace aparecer el punto sobre el piano.

710: Toma la alta frecuencia de cada nota.

720: Toma la baja frecuencia de cada nota, y verifica si hemos pulsado la opción 1 ó la 2.

725: Si hemos pulsado la opción 1 el ordenador espera a que pulsemos una tecla. Mientras no lo hagamos, la nota seguirá sonando.

727: Si hemos pulsado la opción 1 saltará a la línea 745.

730: Si hemos pulsado la opción 2, el ordenador hará sonar la nota mientras TI no sea igual al valor de los números 7 y 8 de cada dato contenido en A\$. A la vez enviará la ejecución del programa a la subrutina 3000 que se encarga del acompañamiento.

745: Pone la envoltura a 0 para que deje de sonar la nota. También borra el punto que había sobre el teclado del piano.

750: Y el bucle se repite para continuar con otra nota.

800-2070: Datos que contienen la información antes mencionada. Estos datos se almacenan en una matriz A\$(i) de 90 elementos y mediante funciones de manipulación de cadenas como habéis visto en las líneas 704 a 730 descompone cada dato de 10 números en: 1 de 3 números, otro de 3 números, 1 de 2 números y otro de 2 números.

3000-3020: Subrutina de acompañamiento.

Y ya nada más. Aunque reconozco que es un palo lo de copiar todos los datos, pensad que más me ha costado a mí encontrar los valores y sinceramente creo que merece la pena. Recibid un saludo.

```
5 POKES3280,1:POKES3281,1
10 PRINT"LEVEL[CLR]5CRSDR]"TAB(7)"[BLK]
EL[SPC]CONC[RT]I[SPC]JEL[SPC]JH[SPC]JUE[
":A=54272
15 POKE1190,112:POKE55462,0:FORL=119
1TO1214:POKEL,64:POKE+R,0:NEXT
20 POKE1,110:POKE+R,0:POKE1255,66:P
OKES5527,0:POKE1295,125:POKES5567,0
25 FORL=1294TO1271STEP-1:POKE1,64:PO
KE+R,0:NEXT:POKE1,109:POKE+R,0
30 POKE1230,66:POKES55502,0
35 PRINT"HOH[BCRSDR]1BCRSR[PORT]SPC]
JOSE[SPC]L.[SPC]JH[BCRSL]1CRSLJ-[CRSDR]
02"
40 PRINT"3CRSDR]SPC]SHIFTB]DESHIFTB]
ESHIFTB]FESHIFTB]GESHIFTB]JESHIFTB]
BCSHIFTB]GESHIFTB]JESHIFTB]JESHIFTB]
GESHIFTB]JESHIFTB]JESHIFTB]JESHIFTB]
GESHIFTB]JESHIFTB]JESHIFTB]JESHIFTB]
GESHIFTB]JESHIFTB]JESHIFTB]JESHIFTB]
GESHIFTB]JESHIFTB]JESHIFTB]JESHIFTB]
2CSHIFTB]JESHIFTB]JESHIFTB]JESHIFTB]
2CSHIFTB]JESHIFTB]JESHIFTB]JESHIFTB]
3CSHIFTB]JESHIFTB]JESHIFTB]JESHIFTB]
4CSHIFTB]JESHIFTB]JESHIFTB]JESHIFTB]
4CSHIFTB]JESHIFTB]JESHIFTB]JESHIFTB]
55 REM DIBUJO PIANO CON POKES*****
*****
*****
```

## FERRE-MORET S.A.

## MAS CERCA DE SU CASA

Encontrará el juego nº 1 de U.S.A. SAUCER ATTACK! en la relación de tiendas de informática o kioscos que detallamos:

#### En BARCELONA

Teléfono Computar, C/. Buenos Aires, 57  
Novo-Digit, S. A., Aragón, 472  
Eletro-Soft, Gran de San Andrés, 128  
Eletro-Soft, Villanar, 104  
Electrónica Viva, Mosen Jacinto Verdader, 78, Sta. Coloma  
de Gramanet  
Radio Deforest, Viladomat, 185  
Eletro-Soft, Gran de San Andrés, 128  
Areni Sams Estación, Santa Ferruca  
La Librería, S. A., Avda. Sarrià, 2  
Librería Artos, Mayor de Sarrià, 2  
Lector Hernández, Reina Elena, 2  
Librería Bosch, Ronda Universidad, 11  
Ediciones Z, Paseo de Gracia, 19  
Mallorca, Rambla Catalana, 86  
Lector y revistas BRUGUERA, Rambla Catalana, 72  
NAPS, Balmes, 244  
Isa del Tesoro, Muntaner Girona, 42

Mandri, Mandri/Cerdanyola  
Calvet, Plaza Nubor de Arco  
Kiosco, Camada Ramblas  
Librería Catalana, Ed. S. Pedro Pto. Corte Inglés  
Kiosco Vespertino, Paseo de Gracia  
Kiosco Druguera, Paseo de Gracia  
Librería Francesa, Paseo de Gracia  
Kiosco La Oca, Plaza Calvo Sotelo  
Kiosco Zanón, Plaza Calvo Sotelo  
Kiosco Francesa, Sofía, Frente Hotel Princess Sofía  
Kiosco Inter, Fontibona, Plaza Manuel Girona  
Kiosco Casa Caspe  
Kiosco Martos, Ramblas  
Kiosco Colln, Ramblas  
Kiosco Cervilla, Bona Diagonal Pto. Corte Inglés  
Kiosco Tele Express, Paseo de Gracia  
Kiosco Sol, Ramblas Plaza Catalana, 13  
Kiosco Via, Plaza Catalana, 13  
Druguera David, Tuset  
Librería Arcadia, Tuset-Pto. Arcadia  
Kiosco Corte Inglés, Corte Inglés, Diagonal

Vic (Barcelona) Servi Comput. C/. Moragas, 46, Hjs.  
Vilafranca del Penedès, Rambla Computer Center,  
Ctra. de Igualada, 21  
Tarragona, Electrónica RRI, C/. Ramón y Cajal, 64  
Gerona, Represom, S. A., C/. Emilio Grahó, 17 bis  
Lleida, Tilsa, Doctor Fleming, 53  
Noya, Andorra 2, Avda. Bonavall, 8, Pampuna  
Palma de Mallorca, Ada Computer, Paseo Independencia, 24-26  
Bilbao, Ayman, Bna de Otero, 43  
Santia, Bilbao, C/. Via Galdak, 2  
Santiago de Compostela, Tadel, C/. Menzies, 15  
Santiago de Compostela, Hail Software, C/. Alférez  
Provisional, 2, ent. E  
Torrevieja, Cantabria, Informática Siglo XXI,  
C/. San José M. Pereda, 1  
Burgos, Contru Wokomo, Alejandro Rodríguez Valcárcel, 9  
Madrid, ABC Informática, Zarzosa, 91-4-0  
Madrid, Key Informática, Embajadores, 90 tienda  
Las Palmas de Gran Canaria, Pedro González Melán,  
Ramón, 5 Vaguet  
Palma de Mallorca, Dimel, S. A., Juan de Cremona, 4

# ¡POR FIN!

# SAUCER ATTACK!

Ha llegado un N° 1  
en USA para su

Commodore 64

Ud. está en su posición vigilando  
Washington D.C.

La ciudad parece apacible desde  
su puesto de defensa y no hay los  
signos de la invasión Extraterrestre  
que Ud. espera.

De pronto el silencio es alterado  
por el penetrante silbido de los  
platillos volantes. Son los invasores  
descendiendo sobre la ciudad.

Ud. entra en acción y pone en su  
punto de mira al platillo volante.

Aprieta el botón de juego y sale un  
misil que destruye al ovni  
desintegrándose el platillo. Otro  
platillo volante se acerca a los  
edificios y antes de que Ud. se de  
cuenta destruye el monumento de  
Washington C.D. con un rayo de  
energía atómica. La batalla  
transcurre durante el día y en un  
momento determinado pasa a ser  
de noche.

Si un platillo aterriza todo está  
perdido. Cuando Ud. consiga  
neutralizar el ataque el juego  
continuará y se deberá destruir la  
nave madre. Si lo consigue una gran  
celebración le espera cuando  
vuelva a Washington D.C.



An action game  
for the  
Commodore 64™

Joystick required



FOTOS  
TOMADAS DIRECTAMENTE  
DE UN MONITOR 1702  
COMMODORE

**SAUCER ATTACK**  
es realmente  
un JUEGO DISTINTO

## INSUPERABLE CALIDAD AUDIO-VISUAL

PEDIDOS

**FERRE - MORET S.A.**

Tuset n.º 8, entlo. 2.º Tel. 218 02 93  
BARCELONA 08006

**DISTRIBUIDORES  
Y PARTICULARES**

NOMBRE.....  
DIRECCION.....  
CIUDAD..... PROVINCIA.....

- ☐ Adjunto cheque  
☐ Contra reembolso

Gastos envío: 300 ptas.

Precio Venta  
**9.975 ptas.**



```

60 FORL=1585T01623:POKEL,121:POKEL+A
,0:FORC=1T08:NEXT:NEXT
65 L=1625:GOSUB400:L=1627:GOSUB400:F
ORL=1631T01635STEP2:GOSUB400:NEXT:L=
1639
70 GOSUB400:L=1641:GOSUB400:FORL=164
5T01649STEP2:GOSUB400:NEXT:L=1653:G0
SUB400
75 L=1655:GOSUB400:FORL=1659T01663ST
EP2
80 GOSUB400:NEXT
85 FORL=1629T01657STEP14:GOSUB410:NE
XT:L=1637:GOSUB410:L=1651:GOSUB410
90 POKE1945,109:POKE1945+A,0:POKE198
3,125:POKE1983+A,0
95 FORL=1946T01981STEP2:POKEL,64:POK
EL+A,0:POKEL+1,113:POKEL+1+A,0
97 FORC=1T020:NEXT:NEXT
100 POKEL,64:POKEL+A,0
200 PRINT"[9CRSRD][35PC]OCTAVA-2[25PC]
[SHIFTB][25PC]OCTAVA-3[35PC][SHIFTB]
[25PC]OCTAVA-4[40M]" :GOT0450
400 FORK=LTL+120STEP40:POKEK,160:PO
KEK+A,0:FORC=1T08:NEXT:NEXT
402 FORM=KTK+120STEP40:POKEM,93
405 POKEM+A,0:FORM=1T08:NEXT:NEXT:R
ETURN
410 FORK=LTL+200STEP40:POKEK,93:POK
EK+A,0:FORC=1T08:NEXT:NEXT:RETURN
450 REM AJUSTES SONOROS VOCES 1 Y 3#
*****
500 FORL=A TO A+23:POKEL,0:NEXT
510 POKEA+24,15:REM VOLUMEN
520 POKEA+5,25:REM ATAQUE/DECAE VO
Z 1
525 POKEA+19,25:REM ATAQUE/DECAE VO
Z 3
530 POKEA+6,24:REM SOSTEN/ESCAPE V
02 1
535 POKEA+20,27:REM SOSTEN/ESCAPE V
02 3
540 POKEA+3,0,REM PULSO ALTO VO
Z 1
545 POKEA+17,0,REM PULSO ALTO VO
Z 3
550 POKEA+2,255:REM PULSO BAJO VO
Z 1
555 POKEA+16,255:REM PULSO BAJO VO
Z 3
600 DIMA$(89):FORJ=0T089:READ A$(J):
NEXT
610 PRINT"[CHOM][CRSRU]1.-[SPC]MANUAL
:PRINT"2.-[SPC]AUTOMATICO[CRSRU]"
625 GETOP$:IFOP$="1"THENPRINT"[CHOM][CRSRD]
[RVSON]1.-[RVSOFF]" :GOT0700
630 IFOP$="2"THENPRINT"[CHOM][2CRSRD]
[RVSON]2.-[RVSOFF]" :GOT0700
640 GOT0625
700 FOR J=0T089:TI$="000000"

```

```

704 C=1664+VAL(RIGHT$(A$(J),2))
705 POKEC,81 :POKEC+A,0:POKE54276,65
710 POKE A+1,VAL(LEFT$(A$(J),3))
720 POKE A,VAL(MID$(A$(J),4,3)):IFOP
$="2"THEN730
725 GETT$:IFT$=""THEN725
727 GOT0745
730 IFIT<VAL(MID$(A$(J),7,2))THENGOS
UB3000:GOTO 730
745 POKEA+4,0:FORB=1T010:NEXT:POKEC,
32
750 NEXTJ:END
800 REM: ALTA FRECUENCIA,BAJA FRECU
NCIA,DURACION DE CADA NOTA Y POKE EN
LA
810 REM:MEMORIA DE PANTALLA PARA IND
ICAR LA NOTA QUE SE ESTA TOCANDO****
*****
1000 DATA 0102051318,0091591616,0102
056018,0102051418,0111141520
1010 DATA 0122163122,0122161922,0122
162522,0111143220,0102051318
1020 DATA 0091591616,0102056818,0102
051118,0111141220,0122161622
1030 DATA 0141074824,0160474226,0122
162622,0111142020,0111142020
1040 DATA 0102052118,0102051518,0091
591316,0081471614,0091595916
1050 DATA 0102051618,0081471314,0080
231412,0081474614,0091591716
1060 DATA 0080231312,0070531410,0080
237212,0080231012,0081471514
1070 DATA 0091595816,0081471214,0080
231512,0080238112,0070535710
1080 DATA 0070531318,0080231212,0081
471414,0091592816,0102051218
1090 DATA 0111141520,0122161522,0141
071224,0122161322,0141074624
2000 DATA 0141071824,0160472126,0170
372028,0170372028,0170372128
2010 DATA 0160473626,0122161322,0111
141420,0122166622,0160471426
2020 DATA 0160471226,0141071324,0141
072224,0122162922,0111141120
2030 DATA 0122161122,0141074524,0122
160922,0111141020,0102051118
2040 DATA 0111144520,0102051018,0091
590916,0081471314,0091596016
2050 DATA 0081471714,0091591716,0102
051718,0111141720,0122161722
2060 DATA 0111141920,0102057618,0070
532010,0080235152,0081471314
2070 DATA 0091595616,0081471614,0080
231612,0080239912,0070539910
3000 REM SUBROUTINA ACOMPAÑAMIENTO **
****
3005 D=0+1:IFD<VAL(MID$(A$(J),7,2))+
15THENRETURN
3010 POKEA+10,65:POKEA+15,14:POKEA+1
4,107:FORM=1T080 :NEXT D=0
3020 POKEA+10,64:RETURN

```

### !!!Atención!!!

Alfonso Ramírez Ramírez  
David Caballero  
Daniel González Alonso  
S. Castell Martorell

Enviadnos vuestras direcciones  
para que podamos devolverlos las cintas que nos enviadsteis.



## VIC-20 + 3K expansión

### Batalla

Ricardo David Alba Atencia  
c/ Sumatra, Q-2  
El Atabal (MÁLAGA)

Este juego simula una batalla entre dos jugadores, cada uno de ellos dispone de 23 ejércitos y otros

tantos cañones, además de una bandera.

Cada jugador tiene una pantalla, de la que puede salir para enfrentarse con su oponente a través de uno de los tres puentes situados en el lado opuesto de la posición inicial de sus fichas.

Las fichas aparecen alineadas una debajo de la otra, a excepción de la bandera que está aislada.

Los cañones al igual que las banderas no se pueden mover. Los ejércitos pueden avanzar o retroceder un máximo de dos espacios en las direcciones del plano.

Si se desea mover una ficha habrá que posicionar el cursor sobre la ficha a mover mediante las teclas de movimiento de cursor, a continuación se pulsará la barra espaciadora con lo que la ficha quedará seleccionada para su movimiento. A continuación se volverá a mover el cursor hasta la posición de la pantalla en la que deseamos poner nuestra ficha y pulsaremos la tecla "@" con lo que la ficha quedará movida automáticamente.

Si queremos pasar una ficha de una pantalla a otra habrá que posicionarla antes al final del puente, pudiendo colocarse al turno siguiente en el otro lado del puente.

Cuando no se deseen mover todas las fichas, se moverán las que se deseen y a continuación se pulsará la tecla "=" con lo que el turno pasará al otro jugador apareciendo su pantalla.

El programa permite visualizar la pantalla deseada con solo la pulsación de la tecla "-" que visualizará la primera pantalla o la tecla "+" que visualizará la segunda pantalla.

A su vez al pulsar la tecla HOME el cursor se situará en la esquina superior derecha, al pulsar la tecla "E" el cursor se situará en la esquina superior izquierda, al pulsar la tecla "T" se colocará en la esquina inferior derecha, y al pulsar la tecla "A" irá a la esquina inferior izquierda.

Si colocamos una ficha a una distancia de uno, dos o tres espacios de un tanque este dispara contra nosotros, y si obtiene una tirada inferior a cuatro nos habrá destruido, desapareciendo nuestra ficha del terreno de juego.

Un tanque será destruido al posicionar una de nuestras fichas sobre el tanque que deseamos destruir.

Dos fichas entrarán en combate cuando estén juntas, resultando ganadora la ficha que consiga una puntuación mayor que la otra; en caso de empate, la tirada se repetirá.

Cada vez que el dado o los dados aparezcan, el programa se parará hasta que se pulse una tecla cualquiera.

El juego acaba cuando una ficha enemiga se posiciona sobre la bandera del otro jugador resultando este último perdedor de la batalla.

El juego corre en el VIC-20, pero necesita una ampliación de 3K de RAM. Si se dispone del SUPEREXPANDER, habrá que anular la influencia de dicho cartucho en la memoria ROM del ordenador, ya que interfiere en los gráficos del programa destruyéndolos por completo. Esto se consigue con el siguiente programa que debe ser ejecutado antes de cargar el programa BATALLA.GRAF. El programa es el siguiente:

```
POKE 256,120:POKE257,76:POKE258,47:POKE259,253:SYS256
```

El programa está dividido en dos:

- BATALLA.GRAF que crea los gráficos;
- BATALLA.PROG el juego propiamente dicho.

La segunda parte se divide en:

Lineas:

- 1-1000: Inicialización de variables.
- 1020-1097: Introducción de órdenes.
- 1097-1109: Controla el cambio de turno.
- 3000-3040: Cede el turno al otro jugador.
- 4000-4990: Diversos casos de combate y simula dado.
- 5000-5100: Paso de cursor a otra pantalla.
- 6000-6130: Dibuja campo y fichas.
- 6900-7003: Conserva el lugar pisado por el cursor.
- 8000-8080: Elige ficha a mover.
- 8400-8640: Mueve ficha y controla la posible batalla.
- 8900-9070: Mueve el cursor a esquinas.
- 1000: Mira el carácter que hay debajo del cursor.
- 11130-11268: Mensaje de fin.



### LISTADO 1

```
1 POKE36879,8
10 PRINT"[CLR][WHT][3CRSRD][2CRSRR][3SHIFT*]
[SHIFT][SPC][SHIFT][2SHIFT*][SHIFT]
[SPC][SHIFT][2SHIFT*][SHIFT][SPC][SHIFTU]
[2SHIFT*][SHIFT]
20 PRINT"[2CRSRR][SHIFT-][2SPC][SHIFT-]
[SPC][SHIFT-][2SPC][SHIFT-][2SPC][COMM]
[COMM][2SPC][SHIFT-][2SPC][SHIFT-]
30 PRINT"[2CRSRR][SHIFT-][2SPC][SHIFT-]
[SPC][SHIFT-][2SPC][SHIFT-][2SPC][COMM]
[COMM][2SPC][SHIFT-][2SPC][SHIFT-]
40 PRINT"[2CRSRR][3COMM][SHIFTK][SPC]
[SHIFTJ][2SHIFT*][SHIFTK][2SPC][COMM]
[COMM][2SPC][SHIFTJ][2SHIFT*][SHIFTK]
50 PRINT"[2CRSRR][3COMM][SHIFTJ][SPC]
[SHIFTJ][2COMM][SHIFTJ][2SPC][COMM]
[COMM][2SPC][SHIFTJ][2COMM][SHIFTJ]
60 PRINT"[2CRSRR][SHIFT-][2SPC][SHIFT-]
[SPC][SHIFT-][2SPC][SHIFT-][2SPC][COMM]
[COMM][2SPC][SHIFT-][2SPC][SHIFT-]
70 PRINT"[2CRSRR][SHIFT-][2SPC][SHIFT-]
[SPC][SHIFT-][2SPC][SHIFT-][2SPC][COMM]
[COMM][2SPC][SHIFT-][2SPC][SHIFT-]
80 PRINT"[2CRSRR][3SHIFT*][SHIFTK][SPC]
[SHIFTK][2SPC][SHIFTJ][2SPC][COMM][COMM]
[2SPC][SHIFTK][2SPC][SHIFTJ]
```



### Tele Sant Just

Mayor, 2. Tel. (93) 371 70 43  
SAN JUST DESVERN (Barcelona)

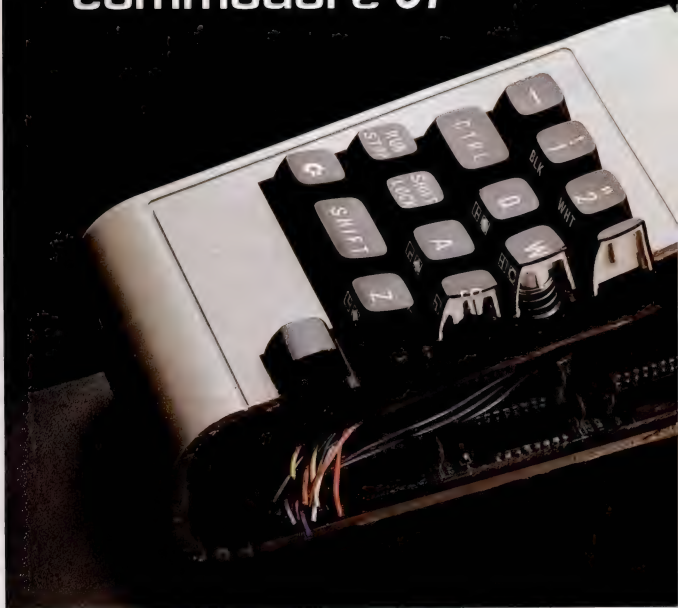
INTERFACE para recibir y transmitir  
CW y RTTY en el VIC-20 y  
COMMODORE 64

INTERFACE para poder conectar cualquier  
cassete a los ordenadores  
COMMODORE 64.

BASE DE DATOS en cassette  
ARCHIVO DE OSL

"NECESITAMOS DISTRIBUIDORES"

# commodore 64



300441

## ¿Utiliza sólo un

Utilizar un COMMODORE 64 únicamente para jugar, es como pedirle a Albert Einstein que nos resuelva la raíz cuadrada de cuatro.

El cerebro del ordenador apenas funciona. Para aprovecharlo al máximo, lo que Vd. necesita son programas interesantes (archivo de documentos, educación interactiva, estimulantes juegos o proceso de textos) y lógicamente los

periféricos adecuados:

- La unidad de disco Commodore, un almacenaje de datos muy rápido, fiable y de gran capacidad.
- La unidad de cassette Commodore, una forma económica de introducir y almacenar programas.
- El monitor en color Commodore, para

MICROELECTRONICA Y CONTROL c/ Valencia, 49-53. 08015 Barcelona. c/ Princesa, 47, 3º G. 28008 Madrid.  
UNICO REPRESENTANTE DE COMMODORE EN ESPAÑA



COMMODORE COMPUTER. Advierte que al comprar uno de sus productos exija la garantía de MICRO-ELECTRONICA Y CONTROL, S.A., única válida para todo el territorio nacional.

# 10% de su cerebro?

- lograr que los textos y gráficos sean más claros y vivos que en un televisor.
- 3 impresoras y un printer-plotter, para copias en papel, en blanco y negro o color.
- Los joystick y paddles, para conseguir que los juegos sean más excitantes.

Sea exigente. Si su cerebro, el de su COMMODORE 64, puede darle 100, no se conforme con 10.



COMMODORE MPS 801  
Impresora de matriz



COMMODORE MCS 801  
Impresora cuatro colores



COMMODORE DPS 1101  
Impresora de margarita



COMMODORE 1520  
Printer plotter cuatro colores

COMMODORE 1541  
Unidad de disco 170 K



COMMODORE 1530  
Unidad de cassette



COMMODORE 1701  
Monitor con color y sonido

JOYSTICKS  
PADDLES  
SOFTWARE Y LIBROS

 **commodore**  
COMPUTER

la emoción del futuro



```

90 PRINT"[3CRSRD][4CRSRR][SHIFT*][SHIFT]
[3SPC][SHIFT*][SHIFT][3SPC][SHIFT]
[2SHIFT*][SHIFT]
100 PRINT"[4CRSRR][SHIFT-][4SPC][SHIFT-]
[4SPC][SHIFT-][2SPC][SHIFT-]
110 PRINT"[4CRSRR][SHIFT-][4SPC][SHIFT-]
[4SPC][SHIFT-][2SPC][SHIFT-]
120 PRINT"[4CRSRR][SHIFT-][4SPC][SHIFT-]
[4SPC][SHIFT-][2SPC][SHIFT-]
130 PRINT"[4CRSRR][SHIFT-][4SPC][SHIFT-]
[4SPC][SHIFT-][2SPC][SHIFT-]
140 PRINT"[4CRSRR][SHIFT-][4SPC][SHIFT-]
[4SPC][SHIFT-][2SPC][SHIFT-]
150 PRINT"[4CRSRR][SHIFT-][4SPC][SHIFT-]
[4SPC][SHIFT-][2SPC][SHIFT-]
160 PRINT"[4CRSRR][SHIFT*][SHIFT*][SHIFT*]
[2SPC][SHIFT*][SHIFT*][SHIFT*][2SPC]
[SHIFT*][2SPC][SHIFT*]
170 PRINT"[2CRSRD][3CRSRR][PUR]POR[SPC]
RICARDO[SPC]ALBA
980 G1=32768:G2=7168
990 FORI=0T0512:K=PEEK(G1+I):POKE(G2
+I),K:NEXTI
1000 DATA0,126,189,219,231,231,219,1
89,126
1002 DATA1,195,189,126,34,34,126,189
,195
1004 DATA0,0,238,238,0,0,187,187,0
1040 DATA3,255,255,255,255,255,255,2
55,255
1042 DATA6,0,224,192,224,240,224,192
,224
1044 DATA7,0,7,3,7,15,7,3,7
1046 DATA2,0,254,238,238,130,238,238
,254
1098 DATA9,255,255,255,255,239,255,
255,255
1100 DATA50,255,255,191,255,255,255,
251,255
1102 DATA51,255,255,191,255,239,255,
251,255
1104 DATA52,255,255,187,255,255,255,
187,255
1106 DATA53,255,255,187,255,239,255,
187,255
1108 DATA54,255,255,171,255,255,255,
171,255
1510 FORI=0T012:READC:FORJ=0T07:READ
K:POKEG2+(C*8)+J,K:NEXTJ:NEXTI
1600 POKE52,28:POKE56,28

```

#### LISTADO 2

```

1 POKE36879,0:R$(1)=[CRSRU][BLU][17CRSRR]
CCCCC":CH(1)=8180:IX=12:Y=X:V1=X:X1=X
:POKE36878,15
2 DIMJ(2,22),C(2,22),M(2,22):CH(2)=0
163:B$(1)=[CRSRU][RED][17CRSRR]HHHH
H":S=7680
5 R$(2)=[BLU]CCCCC":AC=32
411 POKE36869,255
415 C0=30720
417 B$(2)=[RED]HHHHH
420 FORI=0T022:J(1,I)=S+2:I#22:J(2,I)
=-1#J(1,I)+17:C(1,I)=J(1,I)+2:C(2,
I)=J(2,I)+2
421 NEXTI:TU=1
1000 T=1:D=7701:GOSUB6000
1020 GETT$

```

```

1030 IFT$="":THENT=1:GOSUB6000:GOSUB
10000
1040 IFT$="":THENT=2:GOSUB6000:GOSUB
10000
1050 IFT$=[CRSRU]:THENX=X-1:X2=1:GO
T07000
1060 IFT$=[CRSRU]:THENX=X+1:X2=-1:GO
T07000
1070 IFT$=[CRSRL]:THENY=Y-1:Y2=1:GO
T07000
1080 IFT$=[CRSRR]:THENY=Y+1:Y2=-1:GO
T07000
1091 IFT$=[SPC]:THEN8000
1092 IFT$="0":THEN8500
1093 IFT$="E":THENGOSUB9000
1094 IFT$=[HOM]:THENGOSUB9020
1095 IFT$="*":THENGOSUB9040
1096 IFT$="^":THENGOSUB9060
1097 IFT$="":THEN3000
1098 GOSUB5000
1097 POKES+Y+X#22,35
1100 FORI=0T022:IFM(TU,I)=3THENNEXT:
GOTO1103
1101 IFM(TU,I)<2THEN1111
1102 NEXT
1103 FORI=0T022:IFM(TU,I)=3THENNEXT:
GOTO1105
1104 M(TU,I)=0:NEXT
1105 IFTU=2THENTU=1:GOTO1109
1106 TU=2
1109 T=TU:GOSUB6000:GOSUB10000
1111 POKES+Y+X#22,AC
2000 GOTO1020
3000 REM TERMINAR DE
3010 REM MOVER.
3030 FORRE=0T022:IFM(TU,RE)=3THENNEXT
T:GOTO1020
3040 M(TU,RE)=2:NEXT:GOTO1020
4000 REM BATALLA.
4010 IFT=1THEND(1)=7720:D(2)=7722:GO
T04030
4020 D(1)=7703:D(2)=7705
4030 LK=2:GOSUB4000
4060 IFQ(1)=Q(2)THEN4030
4070 IFQ(1)>Q(2)THENJ(T2,I)=0:M(T2,I)
=3:GOTO4090
4090 J(TU,FI)=0:M(TU,FI)=3
4090 GOSUB6000:GOSUB10000:GOTO0620
4100 IFTU=1THEND(1)=7704:GOTO4120
4111 D(1)=7721
4120 LK=1:GOSUB4000
4130 IFQ(1)<4THENJ(TU,FI)=0:M(TU,FI)
=3
4140 SC=1:GOSUB6000:GOSUB10000:GOTO0
620
4800 FORTD=1TOLK
4810 FORG=0T010:Q=INT(RND(1)*6)+1:P0
KE36876,128+Q:POKE(DT),48+Q:POKE(T
D)+C0,1
4820 GETT$:IFT$=""THENNEXT
4830 POKE36876,0:Q(1)=INT(RND(1)*6)
+1:POKE(DT),48+Q(1):NEXT
4835 GETT$:IFT$=""THEN4835
4840 RETURN
4850 IFTU=1THENT2=2:GOTO4855
4851 T2=1
4855 FORI=0T022:IFABS(C(T2,I))=S+Y+X
#22THEN4870
4866 NEXT
4870 C(T2,I)=0
4890 GOSUB6000:GOSUB10000:RETURN
4900 IF-M=P-21THEN4900
4910 IF-M=P-1THEN4900
4920 IF-M=P-43THEN4900
4940 GOT08639
5000 REM PASA A OTRA

```



# DIRECTORIO



**ELECTRONICA**  
SANDOVAL S.A.

DISTRIBUIDORES DE:

ORIC-1  
CASIO FP200  
ROCKWELL-AIM-65  
VIDEO GENIE-EG-2000  
CASIO FX-9000P  
SINCLAIR SPECTRUM  
OSBORNE 1  
DRAGON-32  
NEW BRAIN  
EPSON HX-20

C/SANDOVAL, 3, 4, 6 MADRID 28010  
TELEFONOS 445 18 33/18 70  
TELEX 47784 SAVL-E

## CENTRO DE INFORMATICA

Las Rozas - Majadahonda  
EMPEZAMOS  
Cursos en BASIC  
cada 15 días  
Directamente en ordenadores  
VIC-20, COMMODORE 64  
SPECTRAVIDEO  
Teléfono: 637 31 51

**FERRE-MORET S.R.**

C/ Tusset, 8 - Entresuelo 2.º  
Tel.: 218 02 93  
08006 BARCELONA



## HACEMOS FACIL LA INFORMATICA

- SINCLAIR • SPECTRAVIDEO
- COMMODORE • DRAGON
- AMSTRAD • APPLE
- SPERRY UNIVAC

Madrid (cuartel) b.d.  
Tel: 253 94 54  
28003 MADRID

Colombia, 39 41  
Tel: 458 61 71  
28016 MADRID

José Ortega y Gasset, 21  
Tel: 411 28 50  
28006 MADRID

Padre Damián, 18  
Tel: 259 86 13  
28036 MADRID

Esmeralda, 100  
Tel: 211 23 62  
28004 MADRID

Avenida Gaudí, 15  
Tel: 296 19 14  
08015 BARCELONA

La Puente, 27  
Tel: 43 68 65  
40002 SEGOVIA

Moisés, 2  
Tel: 601 70 36  
40100 ALBACETE

**casa de software s.a.**

c/ Aragón, 272, 8º, 6.º  
tel. 215 69 52 - 08007 barcelona

- Software profesional para C-64
- Con distribución productos DIGITAL RESEARCH

## RADIO WATT

- ORDENADORES PERSONALES
- ACCESORIOS INFORMATICA
- COMPONENTES ELECTRONICOS
- TELECOMUNICACIONES

Paseo de Gracia 126-130  
Tel. 237 11 82. 08008 BARCELONA

**REM**

- Ordenadores personales Hard y Soft.
- Cursos de Basic.

Oficinas: RENOVACION EN MARCHA, S.A.  
c/ Esmeralda 134 2º int. MADRID-3  
Teléfono (91) 441 24 78  
Temas: REM SHOP 1  
C. Galileo, 4 - MADRID-15  
Teléfono (91) 445 28 08

**MICROS GARDEN S.R.**

- ORDENADORES PERSONALES -

- SINCLAIR SPECTRUM Y PLUS
- COMMODORE 64
- ATARI 800 XL
- ORIC ATMOS
- DRAGON 32 y 64

- PERIFERICOS Y ACCESORIOS
- SOFTWARE PARA TODAS LAS MARCAS
- CURSOS DE BASIC PARA PRINCIPIANTES (PRACTICAS CON ORDENADOR)
- GRAN SURTIDO EN LIBROS Y REVISTAS

Francisco Silvela, 19  
Teléfono 401 07 27. 28028 MADRID

## ORDENADORES

PERSONALES/PROFESIONALES

RED DE ORDENADORES  
PARA COLEGIOS.



**MICROINFORMATICA**

Equipos/ Servicios  
Centro de formación

General Ricardos, 15.  
Tel. (974) 31 14 39. BARBASTRO (Huesca)

## COMPUTERWORLD

es miembro de CW Communications/Inc., el grupo editorial más grande y de publicaciones relacionadas con los ordenadores en 18 países, dando como resultado la cantidad de nueve millones de lectores que leen una o más de nuestras publicaciones.

¡¡¡Suscríbete!!!  
Tel. (91) 231 23 88/95

## MicroSistemas

Todos los meses  
publicaremos  
programas para los  
Commodore.

¡¡Suscríbete!!

Tel. (91) 231 23 88/95



```

5010 REM PANTALLA.
5030 IFV>21ANDT=2THENV=21
5040 IFV>21ANDT=1THENV=0:T=2:GOTO510
5050 IFV<0ANDT=2THENV=21:T=1:GOTO510
5060 IFV<0ANDT=1THENV=0
5090 RETURN
5100 GOSUB6000:GOSUB1000:RETURN
6000 REM DIBUJA CAMPO
6001 REM V FICHAS.
6020 PRINT"[CLR]";
6030 PRINT"[HOM]";:FORH=1TO3:FORI=1T
05:PRINTA*(T):NEXT
6040 FORI=1TO2:PRINTB*(T):NEXT:NEXT
6050 PRINTA*(T):[HOM]
6060 FORI=1TO5:POKECH*(T)+I,3:POKECH(
T)+I+CO,6:NEXT
6065 IFT=2THEN6100
6070 POKE7900,2:FORI=0TO22:IFJ(1,I)>
0THENPOKEJ(1,I),0:POKEJ(1,I)+CO,7
6080 IFC(1,I)<0THENPOKEC(1,I),6:POK
EC(1,I)+CO,5
6081 IFJ(2,I)>0THENPOKEJ(2,I),1:POKE
J(2,I)+CO,3
6090 NEXT:I:RETURN
6100 POKE7921,2:FORI=0TO22:IFJ(2,I)<
0THENPOKEABS(J(2,I)),1:POKEABS(J(2,I
))+CO,3
6110 IFC(2,I)<0THENPOKEABS(C(2,I)),
7:POKEABS(C(2,I))+CO,4
6120 IFJ(1,I)<0THENPOKEABS(J(1,I)),0
:POKEABS(J(1,I))+CO,7
6130 NEXT:RETURN
6900 REM CONSERVAR
6901 REMLUGAR PISADO.
7000 A2=PEEK(S+Y+X*22)
7001 IFA2=30RX<00RX<22THENX=X+X2:I=Y+
Y2:GOTO7004
7002 IFV=22ORY=-1THEN7004
7003 AC=A2
7004 POKE36876,200:POKE36876,0:X2=0:
Y2=0:GOTO1001
8000 REM ELEJIR FICHA
8010 REM A MOVER.
8012 IFT=2THEN8050
8020 FORFI=0TO22:IFJ(TU,FI)=S+Y+X*22
THEN8035
8030 NEXT:FI=-1:GOTO1020
8035 IFM(TU,FI)=0THENM(TU,FI)=1
8040 GOTO1020
8050 FORFI=0TO22:IF-1*(J(TU,FI))=S+Y
+X*22THEN8070
8060 NEXT:FI=-1:GOTO1020
8070 IFM(TU,FI)=0THENM(TU,FI)=1
8080 GOTO1020
8400 REM MOVER FICHA.
8500 IFFI=-1THEN1020
8501 IFM(TU,FI)=20RM(TU,FI)=30RM(TU,
FI)=0THEN1020
8509 V3=(ABS(J(TU,FI)))-S:X3=0
8510 IFY3<22THEN8541
8520 V3=Y3-22
8530 X3=X3+1
8540 GOTO8510
8541 IFV=0ANDT=2ANDTU=1ANDV3=21THENY
3=-1:Y=0:GOTO8560
8542 IFV=1ANDT=1ANDTU=2ANDV3=0THENY
=21:Y3=22:GOTO8560
8543 IFV=1ANDT=1ANDTU=1ANDV3=0THENY
=21:Y3=22:GOTO8560

```

```

8544 IFV=0ANDT=2ANDTU=2ANDV3=21THENY
3=-1:Y=0:GOTO8560
8545 IFV<2ANDTU=1ANDT=1THEN1020
8546 IFV>19ANDTU=2ANDT=2THEN1020
8547 QW=J(TU,FI)
8548 IFQW<0ANDT=1THEN1020
8549 IFQW<0ANDT=2THEN1020
8560 IFX3>X+20RX3<X-20RY3>Y+20RY3<Y-
2THEN1020
8562 IFPEEK(S+Y+X*22)=2ANDTU<0THEN1
1000
8563 IFPEEK(S+Y+X*22)=6ANDTU=20RPEEK
(S+Y+X*22)=7ANDTU=1THEN8050
8569 IFPEEK(S+Y+X*22)=8THEN8580
8570 IFPEEK(S+Y+X*22)<32THEN1020
8580 POKERBS(J(TU,FI)),32
8590 POKES+Y+X*22,TU-1:POKECO+S+Y+X*
22,1:AC=TU-1:X3=0
8591 VA=1:IFT=2THENVA=-1
8592 IFTU=2THENT=1:GOTO8600
8593 T2=2
8600 J(TU,FI)=S+Y+X*22:M(TU,FI)=2
8610 J(TU,FI)=J(TU,FI)*VA
8619 SC=0
8620 P=J(TU,FI):FORI=0TO22:M=J(T2,I)
:C=C(T2,I)
8630 IFP=M+10RP=M-10RP=M+220RP=M-220
RP=M+230RP=M-230RP=M-1010RP=M+21THEN4
010
8631 IFTU=1THEN8635
8632 IFSC=1THEN8635
8633 IFP=C+10RP=C+20RP=C+3THEN4100
8635 IFV=21ANDT=10RY=0ANDT=2THEN4900

```

```

8639 NEXT:GOTO1020
9000 IFT=2THENX=0:Y=5:GOTO10000
9010 X=0:Y=0:GOTO10000
9020 IFT=2THENX=0:Y=21:GOTO10000
9030 X=0:Y=16:GOTO10000
9040 IFT=2THENX=22:Y=5:GOTO10000
9040 X=22:Y=0:GOTO10000
9060 IFT=2THENX=22:Y=21:GOTO10000
9070 X=22:Y=16:GOTO10000
10000 AC=PEEK(S+Y+X*22):RETURN
11000 RENFIN.
11010 REM
11020 FORI=128TO255:POKE36876,I:NEXT
:IFT=255TO127STEP-1:POKE36876,I:NEX
T
11230 POKE36869,240:PRINT"[CLR]";[4CRSRR]
[0SPC]A[SPC]INCSPC]A[SPC]DCSPC]0[SPC]
R
11240 PRINT"[5CRSRD]";[CRSRR]JUGADOR:;[SPC]
"U
11250 PRINT"[3CRSRD]";[CRSRR]OTRA[SPC]
PARTIDA?
11260 GETT$:IFT$=""THEN11260
11268 IFT$="S"THENRHN

```



# COMENTARIOS MODORE WORLD

## Programación basic para microcomputadoras

Autor: Luis Joyanes Aguilar  
Nº pág.: 440.  
P.V.P.: 1.490 Ptas.  
Editorial McGraw Hill España  
C/ Santa Beatriz, 4  
Madrid  
Teléfono: 91/433 87 77

Este libro va destinado tanto para aquellos que comienzan en el mundo de la informática, como para los que ya llevan algún tiempo y desean conocer un poco más a fondo las peculiaridades del basic en los microordenadores más populares del mercado nacional. El autor recomienda a los principiantes que lean detenidamente cada capítulo y practiquen sobre él, antes de pasar al siguiente.

En cada capítulo presenta una serie de ejercicios para que el lector los resuelva sobre un cuaderno o sobre el mismo ordenador, comprobando las aplicaciones del basic estudiadas en cada capítulo y verificando posteriormente los resultados con los que aparecen en el libro.

El capítulo cero del libro está dirigido a los que llegan ahora a la informática. Hace una "microhistoria" del desarrollo de los ordenadores desde su comienzo hasta nuestros días, y expone de modo sencillo los conceptos fundamentales: caracteres, datos, clasificación de las computadoras, hard y soft, el concepto de programa y la arquitectura y funcionamiento de un ordenador (CPU, ALU, entradas/salidas, RAM, ROM, impresoras, cintas, discos, etc.).

El capítulo uno enseña a manejar el ordenador, tanto en modo directo como en modo de programa, estudia las sentencias fundamentales (LET, PRINT, etc.) como editar, ejecutar, modificar y corregir un programa. También indica cómo guardar y cargar programas para volverlos a utilizar posteriormente.

El capítulo dos y siguientes ya explican más detalladamente y a un nivel superior la utilización de diagramas de flujo, variables, sentencias iniciales (input, print, rem), sentencias de control (goto, if, then, for, next, on goto), manejo de caracteres y funciones de cadenas (len, val, left\$, right\$, space\$, etc.), funciones generales (max, min, int, etc.), trigonométricas, definidas por el usuario, entradas y salidas de datos (read/data/restore, spe, tab, pos, print at, print using).

A partir del capítulo 8, el libro sube su nivel, comienza el estudio de la utilización de listas y tablas, dimensionado de matrices (dim), option base, mat print y mat input. También estudia las subrutinas e instrucciones de llamada y retorno. En el capítulo 10 encontramos el estudio de los Peeks y los Pokes, call, usr, sys, memfree, ti\$, y otras funciones especiales (var, wait, swap) y las órdenes de control y depuración de un programa (on error goto y trace entre otras).

Una vez terminado el estudio del basic y su utilización, en el capítulo 11 amplía las nociones que dio en el primer capítulo para detectar errores, modificar programas, numeración de líneas de los programas, etc., con una serie de detalles que si los hubiese intentado explicar anteriormente no los habrían entendido una gran parte de los lectores.

En definitiva este libro es un buen curso de basic o manual de consulta para todos, están bien desarrollados todos sus capítulos y son fácilmente comprensibles siempre que se tengan los conocimientos necesarios para cada uno de ellos. Por esto es importante, en caso de no estar seguro de saber algo, empezar el estudio desde los primeros capítulos, y quien ya tenga conocimientos puede pasar directamente a capítulos superiores.

Uno de los detalles que hacen más interesantes a este libro, es que no se limita a estudiar el Basic del VIC o del C-64, todas las explicaciones son válidas para el basic en general, por lo que sigue siendo útil aunque cambien de ordenador. Además, en sus apéndices, aparecen el juego de caracteres ASCII y las variantes utilizadas en los ordenadores personales más populares del mercado español (IBM/PC, PET/CBM, APPLE II, ZX-81, SPECTRUM, VIC-20/C-64) y el de las impresoras SHEIKOSA.

Otro de los apéndices detalla las características de las impresoras más utilizadas actualmente (EPSON y SHEIKOSA). El siguiente repite las típicas y útiles tablas de



conversion decimal-hexadecimal (que nunca vienen mal), y por último un detalle interesante: un apéndice describiendo los comandos y funciones de cada uno de los ordenadores que citó anteriormente. Esto nos permite poder adaptar a nuestro ordenador algunos programas que consideramos interesantes y que aparezcan publicados para otros ordenadores de diferentes marcas, ya que en la descripción de cada función podemos ver lo que hace e imaginar el modo en que podríamos simular la función de ese comando en nuestro ordenador con otro comando o conjunto de comandos equivalentes. Estos apéndices son la parte más útil de libro para mi, y no me considero un principiante en informática...

## Commodore 64. Qué es, para qué sirve y cómo se usa

Autores: D. Ellershaw/P. Schofield  
Traductor: Ramos Tapia.  
Nº Págs.: 160.  
P.V.P.: 950 Ptas.  
Editorial Norel  
C/ San Gervasio de Cassolas, 79  
08022 BARCELONA

### Una buena ayuda para principiantes

Esta obra va dirigida para aquellos usuarios del ordenador que acaban de comprarlo y se encuentran desconcertados. Quizás estas novedades los chicos pidieran a SS.MM. los Reyes Magos un ordenador "por que su amigo tenía uno...". o en el colegio lo están empezando a utilizar o, es más barato a la larga que los video-juegos, y además papá quiere saber qué es "eso" e intentar llevar una pequeña contabilidad o fichero, etc. El caso es que muchas veces los usuarios se encuentran con un ordenador en su poder y no saben cómo usarlo ya que el manual es demasiado complicado y no aclara muchas dudas.



Para todos estos propietarios del Commodore 64 va destinado este libro, del que su autor dice en el primer capítulo (introducción): "Si encuentra usted el manual difícil de seguir, si ve que comprende solamente las primeras líneas y poco más, entonces ESTE ES SU LIBRO". Estamos totalmente de acuerdo con el autor en esta afirmación, ya que aunque parezca mentira hemos recibido varias decenas de llamadas de personas que han impuesto su dinero a plazo fijo en algún banco, les han dado a cambio el ordenador, y no sabían ni siquiera cómo conectarlo para utilizarlo.

El libro, que originalmente se titula "Very Basic Basic" (Basic muy básico), está bien traducido. Además una nota curiosa: cuesta menos la versión en español que la original inglesa, al menos el ejemplar que hemos encontrado a la venta en uno de los grandes almacenes de la capital.

Esta obra ha sido dividida en cuatro secciones, la primera de ella titulada "antes de programar", como podéis suponer da una serie de consejos, desde cómo conectar el aparato hasta algunos programas simples, pero su función principal es la de familiarizar al lector con los comandos basic más sencillos y cómo utilizarlos en los programas (LET, PRINT, etc.).

En la segunda sección trata el manejo de las variables y las técnicas de programación desde un punto de vista elemental, claro y conciso. La utilización de las variables, bucles, contadores y bifurcaciones han sido ilustradas con numerosos ejemplos que afortunadamente no son sólo teóricos, sino que tienen aplicaciones prácticas, por ejemplo: en el capítulo de manejo de cadenas, para ilustrar el modo de utilizarlas, encontramos un "procesador de textos primitivo" que ocupa solamente seis líneas de programa.

La tercera sección está dedicada al manejo del grabador de cinta (datasette) y la impresora, son sólo dos cortos capítulos con una serie de instrucciones y consejos útiles para el manejo y conservación de estos dos dispositivos.

(Pasa a pág. 72)

# Magia

La **MAGIA** son trucos, la **MAGIA** es divertida.

La **MAGIA** es hacer lo que nadie se ha atrevido y resulta ser la fuente más completa de información para la informática práctica.

La **MAGIA** es una sección mensual llena de consejos, trucos, de esto y aquello del mundo del software, hardware y aplicaciones, trucos descubiertos por los demás que hacen que la informática sea más fácil, más divertida o más animada.

**MAGIA** habla de ideas sencillas, programas de una sola línea, subrutinas útiles, hechos de informática poco conocidos y otras cosas de interés.

Los trucos de magia enviánnoslos comprobados, pues hay varios incorrectos. ¡Ah! y no nos mandéis trucos repetidos... ¡Listos!. Premios: Ver pág. 5.

## EL BUFFER DEL TECLADO

Básicamente, lo que hace es anular el buffer del teclado, que puede ser, en algunas ocasiones, muy engorroso. Voy a ponerlos un ejemplo

```
"mu" clarito:
10 FOR N=0 TO 499
20 PRINT N
30 NEXT N
40 INPUT AS
```

¡¡Qué programas más bonitos me salen!! No, en serio. Cuando corramos este programa, mientras se esté ejecutando el bucle (que se puede ver por dónde va gracias a la línea 20), nosotros teclearemos una palabra, por ejemplo "MI CASA", (por eso del E.T.) y cuando se termine el bucle, ¿qué pasa?, que nos sale el interrogante del INPUT, seguido de la palabra ¡¡MI CASA!! Sobre todo, ¡¡Antes de que aparezca el interrogante!!

Pues no quiero ni contarlos lo que pasa si sustituimos la línea 40 por GET AS: IF AS="" THEN 40.

Esto resulta verdaderamente incómodo cuando se tiene un programa activado por MENU y con sentencias GETs, como, por ejemplo, el programa de AGENDA que os mandé (y del que no sé qué vais a hacer con él, pobrecito).

Todo esto se puede evitar con un sencillo truco, que es ¡anular el teclado! con los POKES 649.

Si ponemos estas nuevas líneas:

```
5 POKE 649, 0
```

```
35 POKE 649, 0
```

Se elimina ese molesto inconveniente. Probad ahora. Si queréis divertirnos, cuando el contador del bucle vaya por el número 250, pulsad la tecla RUN/STOP.

Es para el Commodore 64.

Juan Rafael Oscar Martín y Mihalici  
C/Guayadeque, 4-3º Izda.  
35009 Las Palmas de G. Canaria.

## DESATASCADOR

Os envío este comando DESATASCADOR por si os puede solucionar algún problema, para el Commodore 64.

Ocurre algunas veces trabajando con el ordenador que perdemos totalmente el control del mismo y la pantalla empieza a hacer cosas raras. Pulsamos entonces la tecla RESET (Commodore World nº 3. Pág. 38) muy recomendable para estos casos.



## VELOCIDAD EN LISTADOS

Hola Commodorianos. Esta es la primera vez que escribo a una revista, y os envío un pequeño truco.

¿Os molesta la lentitud con que va nuestro C.T.R.L.? ¿cuando listando un programa lo presionáis no os molesta su lentitud? Pues ahí va el truco, apretando el C.T.R.L., apretar a la vez cualquiera de estas teclas, "C=", "RUN STOP", "1", "2", "Q". El listado pasará rápido y cuando soltéis la segunda tecla continuaréis con la lentitud de C.T.R.L. Y una pregunta, ¿por qué al encender mi 64 y poner PRINT FRE (8) me sale —26627 de memoria libre?

Alonso Soler Mellado  
C/Miguel Reverter, 9-1º-3º  
San Justo Devern (Barná)

NOTA: La respuesta es que cuando conectas el 64, este tiene más de 32K libres, e interpreta esta cantidad como negativa. Para saber cuál es la memoria verdadera, teclea:

```
PRINT 2116-fre(0).
```

## UNA SOLA LINEA

leyendo la sección de Magia de vuestra revista me ha animado a mandaros un par de PROGRAMAS DE UNA SOLA LINEA no tienen utilidad práctica pero, al menos el primero creo que es algo didáctico para aprender los POKES de activación de sonido a la vez que produce un efecto de pasos con eco, esta línea debe de introducirse usando las abreviaturas de los comandos que intervienen en ella: SIN ABREVIAR:

```
1V=54296:fori=35TO0step-3:pokeV, 1x2: poke V-2,235: poke V-1, 1x7: poke V-594: poke V-6,230: next: gotol
```

En cuanto a la segunda no precisa ni mucho menos de los 80 caracteres pero el efecto visual conseguido en la pantalla del monitor es divertido:

```
2 PRINT "[SHIFT-CLEAR]": FORI=1TO255:POKE53281,I: POKE 53280, I: NEXT: RUN
```

La he numerado con un dos esta segunda línea porque si las introducimos las dos en el CBM-64, y eliminamos el GOTOL de la revista se produce un nuevo efecto, es más, si manejamos únicamente la primera línea y variamos el valor segundo de los POKES se pueden crear nuevas pautas de sonido.

Arcadio Segura Arnaú  
C/San Miguel, 61  
La Centa (Tarragona)



# El Misterio de los Pokes (IX)

Por Diego ROMERO



**E**n este número vamos a ver algunas de las posiciones más importantes del mapa de memoria de los "hermanos mayores", es decir, los equipos Commodore de la gama alta. También podemos conocer la manera de pasar programas de un VIC a un PET 3000.

Los punteros que nosotros encontramos en el VIC y el C64 en las direcciones \$2B, \$2C, \$2D, etc., son los que indican el principio y final de la memoria ocupada por programas Basic, principio y final de variables, etc. Todos ellos ya los he explicado en números anteriores, y para encontrar los equivalentes publicamos esta tabla con la dirección correspondiente del C-64 y su equivalente en el PET-CBM 3000.

VIC	PET
\$2B-2C	\$2B-29
\$2D-2E	\$2A-2B
\$2F-30	\$2C-2D
\$31-32	\$2E-2F
\$33-34	\$30-31
\$35-36	\$32-33
\$37-38	\$34-35

Creo que estas son las posiciones más importantes a la hora de intercambiar programas, ya que la única dificultad que se nos presenta al cargar un programa de cinta de un VIC en un PET 3000 es que el VIC carga normalmente los programas en la primera posición de RAM que tenga libre, excepto que el indicador de carga o dirección secundaria utilizada al hacer save sea un 1, con lo que al hacer LOAD la carga se realiza en las mismas direcciones de donde salió el programa, igual que si hiciésemos LOAD""", 1.

Para poder intercambiar los programas en cinta con un PET debemos añadir una línea 0 REM en el PET al principio del programa, luego hacemos LOAD""", 1 para cargar un programa grabado por un VIC, y luego hacemos POKE 1025, 1: POKE 1026, 16. Con estos pokes alteramos el enlace de la primera línea Basic (el 0 REM) a la posición donde se encuentra la primera línea del programa del VIC, luego podemos hacer LIST y si queremos eliminamos la línea cero teclando un cero y pulsando Return. La única condición para que esto funcione es que el programa del VIC 20 no contenga la línea cero y que sea para el VIC 20 sin ampliación. En caso de disponer de ampliación mayor de 3K, el poke a 1026 sería: POKE 1026, 18. Si el VIC estaba ampliado sólo con los 3 K de RAM, el programa cargará perfectamente sin necesidad de teclear la línea cero y los pokes.

Los programas del VIC se pueden utilizar

en los PETs 3000 mientras no empleen instrucciones POKE a la memoria de pantalla, y prescindiendo de los pokes de sonido. La pantalla en el PET 3000 se encuentra a partir de la dirección 32768 (\$8000), mientras que en el VIC 20 no ampliado o sólo con la ampliación de tres K RAM está situada en la dirección 7680 (\$1ED0), y si está ampliado con más memoria se coloca en las direcciones 4096 (\$1000) y siguientes. Si deseáis, sabiendo la diferencia entre las posiciones que ocupan la memoria de pantalla en uno y otro ordenador, podéis utilizar en los PETs los programas que publicamos para el VIC con las modificaciones que necesite.

Para aquellos que dispongan de un CBM 700, les puede ser útil el programa ARRIBA PERISCOPIO publicado en el número 2. Este programa era un desensamblador y un pequeño monitor de lenguaje máquina, y funciona perfectamente en el 700 con sólo añadir una línea al principio del programa en la que especificuemos el banco de memoria en el que vamos a trabajar, puede ser: 500 BANK 15, o hacer un INPUT para escogerlo y luego la instrucción BANK.

Como información adicional para los propietarios del 700, que todavía no lo sepan, la CPU que utiliza es la 6509, que como el 6502, y 6510 puede direccionar solamente 64 K de memoria, pero el 6509 incorpora dos registros en las direcciones 0 y 1 que le permite utilizar la técnica de bancos de direccionamiento, con la que puede acceder a 16 de ellos, cada uno con 64 K, lo que hace un total de 1.048.576 Bytes direccionables (1 Mbyte).

Los bancos se numeran del 0 al 15, y están asignados como sigue:

- Banco 0 reservado para el futuro ???
- Banco 1 reservado para el texto de los programas en Basic.
- Banco 2 almacena los datos (variables, arrays, strings).
- Banco 3 al 14 sirven para ampliaciones de RAM.
- Banco 15 reservado para el sistema (Kernel, DOS, Basic, etc.).

Puede parecer poco el disponer de sólo 64 K de RAM para programas en Basic, pero si tenéis en cuenta que son sólo para el programa, y que las variables se almacenan en otros bancos, podréis ver que es mucha memoria. Como dato curioso, diré que el programa de control de suscriptores que utilizamos en la revista, está escrito en Basic, utiliza ficheros indexados relativos, permite una infinidad de opciones (caducidad, modo de pago, número de cheque, reembolso o talón, etc.) y ocupa un poco menos de 10 K de RAM. Podría ocupar menos de 4 K RAM,

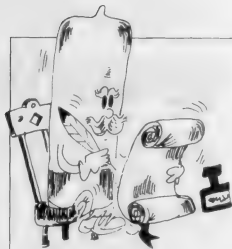
pero le añadí montones de sentencias REM por todas partes para que si alguien quiere modificarlo en el futuro pueda hacerlo sin dificultad.

El banco 15 del 700 es el más interesado, pues en él encontramos todos los dispositivos de entradas/salidas, kernel, etc. Un dato curioso también es que el 700 utiliza el mismo chip generador de sonido que el Commodore 64, por lo que podríamos utilizarlo para aplicaciones musicales como su hermano pequeño, para ello debemos saber dónde se encuentra el SID (MOS 6581) y hacer los pokes correspondientes como los hacemos en el C64. Para que sepáis dónde se encuentra "escondido" el chip, publicamos una lista de direcciones del banco número 15 del 700.

DIRECCION	CONTENIDO
\$0000	Registro de ejecución del 6509.
\$0001	Registro de in dirección del 6509.
\$0002-\$03FF	RAM del sistema, punteros kernel, etc.
\$0400-\$07FF	RAM libre.
\$0800-\$0FFF	RAM utilizada por el sistema
\$1000-\$1FFF	Reservado para los discos integrados.
\$2000-\$7FFF	24 K RAM o ROM de cartucho
\$8000-\$BFFF	ROM intérprete basic 475
\$D000-\$D7FF	RAM de pantalla 80x25.
\$D800-\$DBFF	Controlador de video Motorola 6845.
\$D900-\$D9FF	MOS 6523 TPI para discos integrados.
\$DA00-\$DAFF	MOS 6581 SSI chips de sonido
\$DB00-\$DBFF	MOS 6526 CIA interface para Z80 o 8088
\$DC00-\$DCFF	MOS 6526 CIA
\$DD00-\$DDFF	MOS 6551 ACIA para interface RS-232
\$DE00-\$DEFF	MOS 6525 TPI-1.
\$DF00-\$DFFF	MOS 6525 TPI-2
\$E000-\$EFFF	ROM editor de pantalla.
\$F000-\$FFFF	ROM kernel

Si queréis pasar los programas musicales del 64 al 700 debéis cambiar las direcciones \$D400-\$D7FF al C64 a las direcciones \$DA00-\$DAFF del 700 en el banco 15.

Con estos detalles damos por terminada esta serie y dejamos paso a algunas series más interesantes que nos habéis pedido, la primera de ellas comenzará en el próximo número y es una serie de tres capítulos para un gran sector de nuestro público... los radioaficionados, programas para utilizar el C-64 y VIC-20 en su interesantísimo campo.



## FALLOS DE CALCULO



El programa que mando redondea los resultados con los decimales que queramos. Sin embargo, sucede a veces (muy pocas veces afortunadamente), que los resultados aparecen con todos los decimales. Me pasado bastantes ratos intentando entender por qué sucede esto y no he encontrado respuesta, por lo que me gustaria que resolviese este problema. El método que utilizo para redondear los decimales es el siguiente:

Si deseamos el resultado con 3 decimales y A1 representa un número con 8 decimales, el mismo número con 3 decimales será (A2)

$MM = (10) \div 3$

$A2 = INT(A1MM * .5) / MM$

Rafael Muñoz Alfonso  
C/Costa Brava, 13 (Can Batlle)  
Vallirana (Barcelona)

El método que nos envías es el correcto.

El fallo está en la forma en la que el ordenador realiza las operaciones aritméticas internas, redondeando a veces.

Por ejemplo:

PRINT COS(2°) nos dará 1

Sin embargo, el INT(COS(2°)) ... ¡da cero!

El ordenador ha hallado el coseno de 2° Pi. Para él, es 0,999999999... lo que hace es redondearlo a 1 cuando lo va a imprimir en pantalla. Naturalmente, el INT de 0,99999... es cero. Este tipo de errores es bastante difícil de evitar. Por ejemplo, podíamos hacer:

En vez de: IF INT(COS(2° \* Pi)) = 1 then...

Hacemos:  
IF VAL (STR\$(INT(COS(2° \* Pi)))) = 1 then...

## MUSICA MAESTRO



Me gustaria que me dijeran dónde podría adquirir el libro del Guía del Programador y qué precio tiene.

En el programa publicado en el número 6 MUSICA MAESTRO me gustaria saber si es imprescindible tener impresora o no, de lo contrario ese programa a mi no me funciona. Josep M. Fondevila  
Suscriptor 5130

La Guía del Programador vale unas 3.000 ptas. Puede comprarse en la mayoría de las tiendas especializadas en informática que existen en las ciudades españolas. De todos modos, consulta con Microelectrónica y Control pues ellos podrán darte una mayor información.

Una de las opciones del Programa Musica Maestro era sacar por la impresora los datos de lo que en ese momento estaba en la pantalla. No es imprescindible tener impresora. Si no te funciona será debido a algún fallo en otra zona del Programa.

# CARTA BLANCA

## LENGUAJES PARA EL VIC



Quisiera saber si se disponen de otros lugares de programación para el VIC y de ser así me gustaria saber los precios del mercado.

También me gustaria saber si se ha publicado en España el libro "6502 Programming manual" y en qué editorial.

Ricardo Montañana Gómez, 17 años  
C/Horno n.º 4, 1.º  
Madruguera (Albacete)

Un lenguaje disponible para el VIC-20 es el **FORTH**, pero no tenemos por el momento la información sobre precios, prestaciones, etc. El libro que citas lo hemos visto publicado en España pero en **INGLES**.

## MÁS EDUCATIVOS Y MENOS MARCIANITOS



Quiero decirlos que me gusta vuestra revista, pero que sería conveniente aumentar los programas educativos, pues ya pasamos de marciánitos y cosas por el estilo.

¿Me podréis decir de algún programa sobre música para el VIC sin ampliación?

David Moreno  
C/Lleida, 6, 3  
Sant Joan Despi  
(Barcelona)

Ya hemos contestado a una petición similar a la tuya en este mismo número, el tipo de programas que aparecen en la revista depende en gran parte de lo que los mismos colaboradores nos envíen, de modo que ya sabéis: **MANDARNOS MÁS PROGRAMAS EDUCATIVOS**.

Conocemos algún programa musical con el **VIC SYNTHESIZER**, pero es mejor que pidas información directamente a las casas que distribuyen soft, ellos te enviarán mucha más información.

## ¿VIC + 64 KRAM = C-64?



Soy suscriptor vuestro desde hace 1 año más o menos, y aprovecho esta ocasión para preguntaros 2 cuestiones que quizás

parezcan muy "chorras", pero siempre tengo esta duda y quiero aclararla de una vez por todas.

La primera es: Lleiváis ya 2 ó 3 meses anunciando para el VIC-20 tarjetas de ampliación de memoria. Una es de 64K RAM + 2 K EPROM (En la página 56 del número 9 lo encontraremos). Desearia saber si con esta tarjeta conseguiremos tener un VIC-20 transformado en un C-64, si es así, si los casettes para el C-64 irían bien para mi VIC.

La segunda es cuando nombráis el Super-expander ¿qué es?, un cartucho de ampliación de memoria determinado, o es simplemente el cartucho de 3.5K RAM, de 16K RAM, etc.

Carlos Darder Tobat  
C/Magallanes C'anillos, s/n  
(San Vía)

Palma de Mallorca

El VIC-20 sigue siendo un VIC-20 por mucha memoria que le añadas lo que pasa es que amplias su capacidad de almacenamiento, y con ello el tamaño de los datos y programas que puedes manejar con él, pero nunca se convierte en un C-64 a menos que cambies casi todos los circuitos que lleva dentro (la CPU, el controlador de vídeo, las RAMs, las ROMs, las CIAs, etc.). Por tanto, en general, no

**PONGA UN ORDENADOR EN SU OFICINA O EN SU CASA**

**PROGRAMAS PROFESIONALES CON LOS EQUIPOS:**

**KAYPRO**

OLIVETTI  
PERSONAL  
COMPUTER



*Kent*  
\*\*\*-electronic

DUVICENTRO Duque de la Victoria, 13,  
Tienda 15 Teléf. 39 65 84

podrás cargar los programas de un C-64, en un VIC-20.

**Superexpand** es el nombre que la casa Commodore le dio a un cartucho de ROM que añade comandos gráficos y de sonido al VIC-20, en otros países lo puedes encontrar con una ampliación de memoria de 3K RAM incorporada a sin ella, pero en España normalmente sólo se vende con ella, por tanto siempre que mencionemos el **Superexpand**, ya sabes que es ese cartucho que tiene nombre propio.

## ERRORES EN EVALUACION DE COLEGIOS



Les agradecería que aclararan el por qué el Programa Colegios que venía en el número 5 no funciona. De acuerdo con el

manual del floppy disk Vic-1541, las sentencias

2006, 2007, 2090, 2095, 2710, 2715, 2845, 2860, 2945, 2960, 5120, 5130, 5220, 5230 y 5240 están mal y el programa así lo acusa al ponerlo en marcha. Soy profesor de Enseñanza Media y tengo gran interés en este Programa (el cual, dicho sea de paso, es muy difícil de seguir en su estructura por la gran cantidad de GOTO y GOSUB que tiene).

Fco. Herrera Arrocha  
C/Ruiz de Alda, 36, 2ª. pta. 4  
35007 Las Palmas

La sintaxis de estas líneas es correcta, aunque no coincide con la que pone el manual en español, no da errores, y el programa funciona perfectamente. Si consigues un manual americano y otro alemán, compáralos con el manual inglés y el español que vienen con la unidad de discos, de este modo podrás comprobar que ninguno de ellos coincide en la sintaxis de

muchos comandos, esto es en parte por errores de imprenta, y en otra parte porque la unidad de discos 1541 tiene una gran flexibilidad en la forma de admitir los comandos y ninguna manera es mejor que la otra, ya que todas son equivalentes (al menos que sepamos nosotros, ya que hemos probado todas las formas de los comandos y no apreciamos diferencias).

## CONTABILIDAD EN CINTA



El programa "LLEVA TU PROPIA CONTABILIDAD", ¿se puede adaptar a la cassette, y cómo?

Los datos grabados en una cassette puedo sacarlos en pantalla ¿PERO COMO PUEDO OPERAR CON ELLOS? Gracias.

Haciéndome eco de tus palabras, no se podrían hacer más programas didácticos, formulación de química, teoría de la física, y un largo, etc. Gracias nuevamente y espero que entre todos podamos dar un uso más interesante a los Microordenadores, que aparte de entretenernos nos sirvan para aprender un poco más.

José Manuel Solanas Masip  
C/Labradoras, 6, 3º Dcha.  
Telef. 229 630  
26005 Logroño

El programa "Contabilidad" está hecho para que abra ficheros en DISCO. Hemos tenido problemas con gente que nos enviaba pedidos de CINTAS de este programa. Como son muchas las solicitudes, estamos trabajando en una versión para cinta. De modo que cuando hagáis vuestros pedidos, indicados si queréis la versión que funciona con ficheros en disco o la que funciona con ficheros en cinta.

Por supuesto publicaremos las correspondientes modificaciones para los que prefieren teclear el programa por sí mismos.

Supongo que te refieres a datos grabados como ficheros. El manual explica claramente cómo abrir ficheros para grabar y leer datos.

## EVALUACION DE COLEGIOS



Soy un asiduo comprador de su revista y debo felicitarle por la calidad de la misma. Tengo un Commodore 64 comprado recientemente y por tanto soy nuevo en la micromanía por ello necesito toda la ayuda de quien pueda darme, me animo a escribirle para plantarle tres preguntas:

—En la revista nº 5 editan un programa de evaluación de colegio ¿se puede adaptar a cinta? ¿qué hay que variar?

—¿Cuándo editarán un programa para un reloj de manecilla, he visto algunos en otras revistas pero para otros ordenadores y no sé adaptarlos, espero su contestación, y sobre todo háganla sencilla, para que pueda comprenderla.

Cipriano López Viegas  
Edificio San Jorge, nº 10-2ª  
Badalona (Barcelona).

El programa "Evaluación de Colegios" funciona sólo con disco. Porque usa ficheros de

acceso directo. Como este tipo de ficheros no pueden abrirse en cinta, es realmente difícil adaptarlo.

Un reloj de manecillas salió en el Programa "Es la hora de aprender" del número 7.

## LOAD-ERROR Y SONIDO



Algunas veces después de apagar y encender el ordenador, al hacer un load este me indica: FOUND "XXXX"

## ?LOAD ERROR OUT OF MEMORY

Pero el ordenador todavía no ha empezado a cargar el listado del programa. ¿Cómo puedo solucionarlo?

2) Cuando saco por pantalla el listado de un programa, o mensajes al correr un programa, o realizo dibujos con el Super Expand, el ordenador emite un pitido que va aumentando si colocó más caracteres en la pantalla, sobre todo si los caracteres son en negro, y va disminuyendo si los quito. El Poke de sonido está en 0, y lo hace con varios televisores. ¿Cómo puedo quitarlo? ¿Tiene el VIC-20 algún mando para regularlo como el C-64? (mi ordenador es un VIC-20).

Xavier Naval Marcos  
Paseig Pere III, 80 2ª  
Manresa  
Barcelona

1. Esto se debe a algún fallo en la cinta o en el automat del datasette. En el artículo "Qué hacer con tu datasette para que entres en las cintas rebeldes" que salió publicado en el número 8, puede que encuentres la forma de arreglarlo. 2. A veces suele suceder que no son del todo compatibles el ordenador y la televisión, o que el modulador del ordenador no está bien ajustado. Hay una bobina dentro del modulador que permite solucionarlo, pero antes de intentarlo es mejor que consultes con Microelectrónica y Control.

## BALONCESTO



Su revista es de mi agrado; llevo poco tiempo metido en el mundo de los ordenadores y esta revista me soluciona problemas que

se me presentan.

Querría me contestasen si en España se vende un cartucho de baloncesto, pues sé que en el extranjero ya está a la venta, pero no he oído nada sobre si había llegado a España.

Jesús Pérez

No nos dices qué ordenador tienes, pero sabemos que existe al menos un programa en disco para el C-64 llamado BASKETBALL. Esperamos que el distribuidor nos lo mande para que podamos daros una mayor información a todos.

## CONSULTA SOBRE UNA RESEÑA



Primero quiero felicitarlos por la revista tan buena que hacéis. Quiero preguntaros si los programas de juegos que anunciabais

en el número 4 de la revista para el VIC-20, que son: El camarerero Loco, El superdetective y Hechizado, necesitan periféricos, como pueden ser: Joystick, Ampliaciones de

## VIC 20 TARJETA MULTIFUNCION

- Amplía 64 K RAM
- Zócalo 8 K EPROM
- ¡Traspasa y ejecuta desde cinta programas de cartuchol (Novedad)

p.v.p.: 20.600 pts.

Pedidos a ICR  
C/ Farnés, 39  
Barcelona 08032

Elija forma de pago:

- ☐ Talón bancario
- ☐ Contrareembolso



# PREGUNTONES

memoria o Paddes. También quiero decirlos si podéis publicar más programas de utilidades, educativos, etc. para el VIC-20.

Julán M. Mangiano García  
C/Acher de Canasteros, 16 4º  
18012 Granada

Los programas que citas son para el VIC-20 sin ampliación de memoria, y pueden funcionar con joystick o con teclado. Lo de publicar menos juegos y más utilidades y educativos depende en gran parte de vosotros mismos: si las colaboraciones que nos enviáis son todas juegos, la mayoría de los programas publicados serán juegos, si no enviáis más utilidades, las publicaremos. Todo esto es así por ser esta revista, una revista hecha por los propios usuarios y colaboradores.

## TECLADO MUSICAL



¿Qué instrucción/es es/son necesarias para que al apretar una tecla cualquiera del C64 se produzca un sonido.

Josep Maria Lecuona i Aragones  
c/Sant Miquel, 9  
Sant Feliu de Gíxols  
Gerona

Para conseguir esto, debes tener una rutina en código máquina o en basic en el ordenador que active las notas al pulsar las teclas. En el manual de usuario del C-64 aparece publicada una que se llama piano (o algo parecido)

No existe ningún comando que te permita hacerlo en modo directo.

## INSERT O DELETE



Señores: Después de felicitarlos por el acierto de nuestra revista paso a hacerles una pregunta.

Con mucha frecuencia al pulsar la tecla INS-DEL/esta queda bloqueada sin moverse el cursor hacia la izquierda. Es normal ¿qué hacer en este caso?

Jordi Erra i Fabregat  
Pau Viç, 52  
Sant Cugat del Valles  
Barcelona

La verdad es que no es normal, en todo caso sería así si estuvierais pulsando la tecla SHIFT simultáneamente.

## RELOJ EN PANTALLA



Desearía si fuera posible me dijeran cómo poner en un programa (por ejemplo de juegos) un reloj, mejor dicho, que en un rincón del T.V. mientras estuviera el juego en marcha fuera mostrando la hora, minutos y segundos.

Para eso, entre otras cosas, sirve alguna rutina similar a la rutina en CM que salió publicada en el número 9 en "Curso de Lenguaje Máquina". Si quieres hacer lo mismo en basic, puedes:

- 1) En un lugar por el que pase siempre el programa: PRINT "[HOME]"; TIS
- 2) Hacer una rutina como: 2000 PRINT [HOME]"; TIS y llamarla de vez en cuando. Quizás Diego lo saque en el curso de lenguaje máquina próximamente.

## COLABORACIONES Y ENVÍOS



Benito Santed Pérez  
Pº Pujadas, 17-19-7º 2º  
Barcelona 08018

Los que deseáis: mandarnos programas de colaboraciones, podéis hacerlo en cinta o en disco, este es el mejor método, pero si preferís también podéis mandar los listados por impresora, aunque en este caso se retrasará su publicación algo más.

Las cintas que enviéis se devolverán con algún programa de regalo (de los de la revista, no programas comerciales).

## TINY FORTH



Ivo Pla Vallvé  
c/Capuchinos, 30, 4  
Tarragona

El lenguaje Tiny Forth ¿qué es? ¿Es una versión barata del Forth, o es algo totalmente diferente? Así, es compilado o interpretado.

Sabemos de dos versiones Inglesas del Tiny Forth, una de ellas es una implementación en Basic del lenguaje forth, y permite hacer tus primeros pintos en este lenguaje desde tu ordenador con sólo teclear un pequeño programa que fue publicado en la revista Computing Today, con esto puedes conocer el Forth antes de gastar un montón de dinero en un cartucho que te permita trabajar este lenguaje de verdad.



La otra versión de Tiny Forth, es también inglesa, y es en realidad un cartucho para trabajar en Forth, son solamente 4K ROM, por lo que su vocabulario es muy limitado (normalmente son 16K ROM), pero con el Forth es un lenguaje que te permite definir tu propio vocabulario, es suficiente con disponer de unas pocas primitivas para conseguir crear tu Forth particular.

Como imagino que sabrás, el Forth no es siempre interpretado ni siempre compilado, depende de cómo tú ordenes que se definan las palabras forth. Naturalmente el Tiny forth que está implementado en basic, es una simulación de forth lenta y que trabaja bajo el control del intérprete Basic.

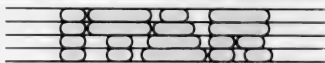
## CATALOGOS



Les ruego me envíen la ficha técnica y lista de capacidades de su ordenador Commodore-64, de la unidad de disco 1541, y la lista de biblioteca de software existente en España.

Pablo del Casar Serrano  
Sor Angela de la Cruz, 10  
Madrid

Publicamos tu carta para que te manden información los distribuidores. Por favor, y esto va para todos: No nos pidáis catálogos, listas de juegos, etc. Nosotros no disponemos de esas listas. Hacélo a las casas de software. Sus direcciones suelen aparecer en los espacios de publicidad.



COMPUTERS, S.A.

PAMPLONA: C/Alfonso el Batallador, 16 (trasera) - Tel. 27 64 04 - Código Postal: 31007  
SAN SEBASTIAN: Plaza de Bilbao, 1 - Tel. 42 62 37 - Télex: 38095-IART - Cód. Post. 20005

¡¡PRECIOS  
ESPECIALES PARA COMERCIANTES!!

COMMODORE - 64  
COMMODORE -16  
ZX SPECTRUM - 48K  
SPECTRUM PLUS  
QL SINCLAIR

PERIFERICOS, PROGRAMAS, LIBROS NACIONALES, EXTRANJEROS, ETC.

6 MESES DE GARANTIA PARA ORDENADORES Y PERIFERICOS

# El duelo

*Es un juego que pone a prueba tus reflejos. Tendrás que reaccionar antes que tu contrincante, pero tampoco te puedes adelantar mucho porque perderás puntos.*



Por David D. BUSCH (RUN, EE.UU.)

Traducido por Valerie SHANKS y adaptado por Alvaro IBÁÑEZ

**T**e apetece poner tus reflejos a prueba contra los del ordenador? No sería muy justo —el C 64 y el VIC-20 son capaces de contar hasta 50 antes de que tú puedas tocar ni una tecla. Los ordenadores son menos inteligentes y, desde luego tienen menos capacidad de movimiento que los seres humanos, pero si les das una tarea sencilla que realizar, lo harán más rápido que cualquiera.

Entonces, con este juego puedes poner tus reflejos a prueba contra los de otro ser humano, es decir realizar un duelo y el ordenador actuará de padrino. Esto parece más razonable, ya que el ordenador no tiene preferencias y podrá calcular la velocidad de respuesta más rápido que tú. Te puedes aprovechar de la velocidad del ordenador sin que ella se aproveche de ti.

En este juego, tú y tu contrincante tenéis que esperar una señal, que indicará que los dos tenéis que intentar a pulsar la tecla que le ha sido asignada. El jugador de la derecha tiene que pulsar la tecla INST/DEL. El que llegue primero se lleva un punto.

Una vez dada la señal, si tú pulsas

cualquier otra tecla no significa ni que pierdas tu turno ni que tengas posibilidades de ganar. Sin embargo, si pulsas cualquier tecla antes de que se dé la señal ¡¡AHORA!! pondrás en marcha una rutina que resta un punto de tu puntuación. El que llegue primero a diez puntos gana el juego.

Los dos jugadores son identificados como el de la izquierda y el de la derecha, y las puntuaciones de cada uno se almacenan en las variables PL y PR, respectivamente. En cada partida, el ordenador elige un tiempo de retraso aleatorio.

En vez de medir los "jiffies", intervalos de 1/60 de segundo, el ordenador cuenta desde uno hasta un número dado, siendo este intervalo de tiempo nuestro tiempo de retraso. Para este juego, el ordenador cuenta al azar una cantidad de números —más de 500, pero menos de 1.500.

Entonces se inicia el bucle For...Next que afecta el tiempo de retraso. Cada vez que se pasa por este bucle, el ordenador comprueba para ver si alguien se adelanta y pulsa una tecla.

**C-64 y VIC-20**

Aunque los ordenadores Commodore pueden hacer uso de varios métodos para realizar esto, uno que no se utiliza es hacer un Peek en una posición de memoria determinada. Lo que hace este juego es aprovecharse de un truco muy popular de programación que resulta bastante más sencillo para que lo pueda comprender el principiante.

Esta técnica es una línea bastante rara de Get A\$. A diferencia de la mayoría de las líneas Get, esta no se repite hasta que el ordenador pulsa una tecla. La forma más común es la siguiente:

```
10 GET A$: IF A$="" GOTO 10
```

Cuando encuentra una línea así, el ordenador busca un solo carácter de teclado y asigna su valor a la variable A\$. Si no se pulsa ninguna tecla, A\$ será igual a nada, o un "string" nulo. Esto también se expresa como dos comillas (" ") que no encierran nada.

Normalmente, el ordenador pasará a la siguiente sentencia del programa. Dado que el C-64 y el VIC-20 repasan el teclado tan de prisa, sería muy difícil llegar a pulsar una tecla justo en el momento en que el ordenador pide una entrada de datos. Por lo tanto, las sentencias Get normalmente se colocan dentro de los bucles, como el que hemos visto antes.

En caso de que A\$ no tiene ningún valor (" "), que sería la mayoría de las veces, se repetirá la línea 10. A\$ tendrá un valor solamente cuando se pulsa una tecla, con el resultado de que el programa pasa a la siguiente línea.

Sin embargo, en el juego "El Duelo", el ordenador no debe hacer una pausa esperando que pulses una tecla. Tiene que seguir contando el tiempo de retraso elegido, y vigilar a la vez que no se pulse ninguna tecla. En vez de dejar que el ordenador se quede colgado en una línea, podrías escribir un módulo como el siguiente.

```
100 FOR N= TO DELAY
```

```
110 GET A$: IF A$<" " GOTO 140
```

```
120 NEXT N
```

```
130 END
```

```
140 PRINT "DISPARO DEMASIADO PRONTO"
```

Con este módulo, el ordenador pasará por el bucle For...Next a una velocidad normal. Si pulsas una tecla al ejecutarse la línea 110, el programa se bifurca a la línea 140, y aparece un mensaje. De no ser así, dado que A\$ es igual a " ", el programa pasará a la línea 120 para repetir el bucle. Ya que cada operación se realiza tan de prisa, parece que el ordenador está repasando el teclado constantemente, aunque está llevando la cuenta del bucle de retraso a la vez.

Esta es la técnica que se utiliza en "El Duelo". Se incluye la línea Get A\$, y si A\$="" ", el programa pasa directamente a Next N.

Sin embargo, si A\$ < " ", el programa pasa a la línea 550, comprueba para ver quién era el culpable y resta un punto de su puntuación. Para que veas, durante el bucle de retraso no se puede pulsar una tecla.

Si el bucle llega a su fin sin ninguna interrupción, la palabra ¡¡AHORA!! aparece en pantalla. Se inicia una señal acústica dentro de un bucle más largo Get A\$. Esto se repite hasta que se pulsa una tecla. El código ASCII de dicha tecla se almacena en la variable A\$.

Si la tecla de flecha a la izquierda o la tecla INST/DEL ha sido pulsada el programa pasa a la rutina de ganador. De no ser así, la tecla se ignora y el programa espera una entrada aceptable.

A medida que tú y tu contrincante van ganando, se van acumulando puntos, y una flecha le apunta al ganador y se sigue jugando. Al final del juego, se anuncia el ganador.

#### VIC-20 SIN EXPANSION

```
10 REM *****
20 REM *
30 REM * EL DUELO *
40 REM *
50 REM *****
60 MH=22
70 POKE36876,15
80 VOICE=36874
90 SO=255
100 PRINT"[CLR]"
110 PRINTTAB(6) "[RVSON][RED]EL[SPC]D
UELO[BLU][2CRSRD]"
120 PRINT"[2SPC]CUANDO[SPC][RVSON]AH
ORA [RVSOFF][2SPC]"
130 PRINT"[CRSRD][SPC]APAREZCA,EL[SPC]
JUGADOR"
135 PRINT"[CRSRD][SPC]DE[SPC]LA[SPC]
140IERDA[SPC]DEBE"
145 PRINT"[CRSRD][SPC]PULSAR[SPC][RVSON]
[RVSOFF].[SPC]Y[SPC]EL[SPC]DE"
145 PRINT"[CRSRD][SPC]LA[SPC]DERECHA
[SPC][RVSON]INST[RVSOFF]"
150 PRINT"[2CRSRD][SPC]EL[SPC]PRIMER
UL[SPC]JUL"
155 PRINT"[CRSRD][SPC]ILLEQUE[SPC]J[SPC]
16[SPC]PUNTOS"
157 PRINT"[CRSRD][SPC]SERIAL[SPC]EL[SPC]
GANADOR!"
160 PRINTTAB(4) "[CRSRD][RVSON][CYN]P
ULSA[SPC]UNA[SPC]TECLA[BLU]"
170 GET# IF#="" THEN170
180 PRINT"[CLR][2CRSRD];TAB(2)"PULS
A[SPC]RETURNA[SPC]PARA"
190 PRINTTAB(2)"SEGUIR."
200 INPUT#
210 PRINT"[CLR]"
220 IFPL>500PR>29THEN500
```

```
230 DELAY=RND(1)*1000+500
240 FORN=1TODELAY
250 GET# IF#<" " THEN550
260 NEXTN
270 PRINT"[3CRSRD];TAB(MA/2-4);"[RVSON]
[BLU]AHORA[BLU]"
280 GET#
290 POKEVOICE,50
300 POKEVOICE,0
310 IF#="" THEN280
320 H=ASC(H#)
330 IFH=500RA=20THEN350
340 GOTO280
350 IFH=95THEN450
360 REM
370 PRINT"[CLR][2CRSRD]"
380 PR=PR+1
390 PRINTTAB(4)"IZQUIERDA:";PL
400 PRINTTAB(4)"DERECHA:";PR
410 PRINTTAB(4) "[2CRSRD]GANADOR[SPC]
[RVSON][RED]--->[RVSOFF][BLU]"
420 A$=""
430 FORN=1T01000:NEXT
440 GOTO180
450 PRINT"[CLR][2CRSRD]"
460 PL=PL+1
470 PRINTTAB(4)"IZQUIERDA:";PL
480 PRINTTAB(4)"DERECHA:";PR
490 PRINT"[2CRSRD][RVSON][RED];TAB(
4)"--->[RVSOFF][BLU][SPC]GANADOR"
500 PRINT"[2CRSRD]"
510 IF#<" " THEN420
520 IF#<" " THEN520
530 FORN=1T01000:NEXT
540 GOTO180
550 H=ASC(H#)
560 IFH=500RA=20GOTO580
570 GOTO280
580 IFH=95THEN580
```

```

590 GOTO630
600 PRINT"[CRSRD][SPC]EL[SPC]JUGADOR
[SPC][RVSON][RED]C---[RVSOFF][BLU]"
PRINT"[CRSRD][SPC]DISPARO[SPC]DEMASI
ADO"
610 PRINT"[CRSRD][SPC]PRONTO!" : PRINT
"[CRSRD][SPC]PIERDE[SPC]JUN[SPC]PUNTO
." : PL=PL-1
620 GOTO660
630 PR=PR+1
640 PRINT"[CRSRD][SPC]EL[SPC]JUGADOR
[SPC][RVSON][RED]C---[RVSOFF][BLU]"
PRINT"[CRSRD][SPC]DISPARO[SPC]DEMASI
ADO"
650 PRINT"[CRSRD][SPC]PRONTO!" : PRINT
"[CRSRD][SPC]PIERDE[SPC]JUN[SPC]PUNTO
." : PR=PR-1
660 PRINTTAB(6)*"[CRSRD][RVSON][CVN]
PULSA[SPC]JUNA[SPC]TECLA[BLU]"
670 GET# : IF#="" THEN670
680 GOTO180
690 PRINT"[CLR][CRSRD]"
700 IFPL>9 THENPRINT"EL[SPC]JUGADOR[SPC]
[RVSON][RED]C---[RVSOFF][BLU][SPC]GHA
NA!" : GOTO720
710 PRINT"EL[SPC]JUGADOR[SPC][RVSON]
[RED]C---[RVSOFF][BLU][SPC]GANA!"
720 PRINT"[CRSRD]"
730 PRINT"OTRA[SPC]PARTIDA?"
740 INPUT#
750 IFLEFT$(B$,1)="" THENRUN

```

#### C-64

```

10 REM *****
20 REM * EL DUELO *
30 REM * EL DUELO *
40 REM * EL DUELO *
50 REM *****
60 NH=40
70 POKE53281,1
100 PRINT"[CLR]"
110 PRINTTAB(17)*"[RVSON][RED]EL[SPC]
DUELO[BLU][CRSRD]"
120 PRINTTAB(8)*"[2SPC]CUANDO[SPC][RVSON]
AHORA[RVSOFF][2SPC]"
130 PRINTTAB(8)*"[CRSRD][SPC]HAREZCA
,EL[SPC]JUGADOR"
135 PRINTTAB(8)*"[CRSRD][SPC]DE[SPC]L
A[SPC]QUIERDA[SPC]DEBE"
140 PRINTTAB(8)*"[CRSRD][SPC]PULSA[SPC]
[RVSON]+[RVSOFF],[SPC]Y[SPC]EL[SPC]D
E"
145 PRINTTAB(8)*"[CRSRD][SPC]LA[SPC]D
ERECHA[SPC][RVSON]INST[RVSOFF]"
150 PRINTTAB(8)*"[CRSRD][SPC]EL[SPC]
PRIMER[SPC]QUE"
155 PRINTTAB(8)*"[CRSRD][SPC]LLEGA[SPC]
A[SPC]10[SPC]PUNTOS"
157 PRINTTAB(8)*"[CRSRD][SPC]SERA[SPC]
EL[SPC]GANADOR!"
160 PRINTTAB(15)*"[CRSRD][RVSON][CVN]
PULSA[SPC]JUNA[SPC]TECLA[BLU]"
170 GET# : IF#="" THEN170
180 PRINT"[CLR][CRSRD];TAB(2)*"PUL
SA[SPC]RETORN[SPC]PARRA"
190 PRINTTAB(2)*"SEGUIR."
200 INPUT#
210 PRINT"[CLR]"
220 IFPL>90RPR>9 THEN690
230 DELAY=RND(1)*1000+500

```

```

240 FORN=1TODELAY
250 GET# : IF#="" THEN550
260 NEXTN
270 PRINT"[CRSRD];TAB(NH/2-4)*"[RVSON]
[BLU]AHORA[BLU]"
280 POKE54296,15
290 POKE54297,1
300 GET#
310 POKE54296,0
320 IF#="" THEN280
330 A=ASC(H$) : IF#950R#20 THEN350
340 GOTO280
350 IF#95 THEN450
360 REM
370 PRINT"[CLR][CRSRD]"
380 PR=PR+1
390 PRINTTAB(4)*"IZQUIERDA:" : PL
400 PRINTTAB(4)*"DERECHA:" : PR
410 PRINTTAB(4)*"[CRSRD]GANADOR[SPC]
[RVSON][RED]C---[RVSOFF][BLU]"
420 H$=""
430 FORN=1T01000 : NEXT
440 GOTO180
450 PRINT"[CLR][CRSRD]"
460 PL=PL+1
470 PRINTTAB(4)*"IZQUIERDA:" : PL
480 PRINTTAB(4)*"DERECHA:" : PR
490 PRINT"[CRSRD][RVSON][RED];TAB(
4)*"C---[RVSOFF][BLU][SPC]JUGADOR"
500 PRINT"[CRSRD]"
510 IF#="" THEN420
520 IF#="" THEN520
530 FORN=1T01000 : NEXT
540 GOTO180
550 A=ASC(H$)
560 IF#950R#200GOTO580
570 GOTO280
580 IF#95 THEN600
590 GOTO630
600 PRINT"[CRSRD][SPC]EL[SPC]JUGADOR
[SPC][RVSON][RED]C---[RVSOFF][BLU]"
PRINT"[CRSRD][SPC]DISPARO[SPC]DEMASI
ADO"
610 PRINT"[CRSRD][SPC]PRONTO!" : PRINT
"[CRSRD][SPC]PIERDE[SPC]JUN[SPC]PUNTO
." : PL=PL-1
620 GOTO660
630 PR=PR-1
640 PRINT"[CRSRD][SPC]EL[SPC]JUGADOR
[SPC][RVSON][RED]C---[RVSOFF][BLU]"
PRINT"[CRSRD][SPC]DISPARO[SPC]DEMASI
ADO"
650 PRINT"[CRSRD][SPC]PRONTO!" : PRINT
"[CRSRD][SPC]PIERDE[SPC]JUN[SPC]PUNTO
." : PR=PR-1
660 PRINTTAB(6)*"[CRSRD][RVSON][CVN]
PULSA[SPC]JUNA[SPC]TECLA[BLU]"
670 GET# : IF#="" THEN670
680 GOTO180
690 PRINT"[CLR][CRSRD]"
700 IFPL>9 THENPRINT"EL[SPC]JUGADOR[SPC]
[RVSON][RED]C---[RVSOFF][BLU][SPC]GHA
NA!" : GOTO720
710 PRINT"EL[SPC]JUGADOR[SPC][RVSON]
[RED]C---[RVSOFF][BLU][SPC]GANA!"
720 PRINT"[CRSRD]"
730 PRINT"OTRA[SPC]PARTIDA?"
740 INPUT#
750 IFLEFT$(B$,1)="" THENRUN

```

WARE



## EXCURSION POR LA MICROPROGRAMACION

# Cursillo de Lenguaje Máquina —IX—

Seguimos con ejemplos prácticos, y en éste número vamos a ver un ejemplo en el que le añadimos nueve comandos para manejo de gráficos al C-64. No he preparado otro programa para el VIC 20 ya que no pretendo publicar programas para los dos por igual, sólo unos "ejemplos", y esta vez le ha tocado al 64, otro día le tocará al VIC. Los que quieran hacer algo parecido en el VIC deberán disponer de alguna ampliación de RAM, no podrán copiar la ROM a RAM por no tener la misma configuración de memoria, pero podrán interceptar la rutina de error del intérprete Basic o cualquier otra rutina para añadir sus comandos, el resto de las ideas y rutinas sirven perfectamente para el VIC (relocalizándolas naturalmente).

Uno de los detalles que hasta el momento no hemos visto y que aparece en el listado del código

Por Diego ROMERO



fueron de este programa es la utilización de los MACROS. Con los macros y un ensamblador que nos permita utilizarlos (como el Macro Assembler de Commodore que distribuye Micro Electrónica y Control), podremos ahorrar tiempo al teclear un procedimiento que vamos a repetir varias veces a lo largo del programa con diferentes parámetros tecleándolo sólo una vez y luego llamando al macro.

La definición de macroinstrucción, nos dice que es una instrucción en lenguaje fuente que es reemplazada por una secuencia de instrucciones escritas en el mismo lenguaje. La macroinstrucción puede especificar valores para los parámetros que deben ser reemplazados en las instrucciones correspondientes. En los manuales de los ensambladores que permiten utilizar macros, vienen explicados más detalladamente. Con palabras sencillas, el macro es una "especie de subrutina", que permite hacer rutinas similares a ella en cualquier punto de un programa, pero la subrutina generada es distinta según los parámetros que se utilizan al llamar al macro.

*A continuación paso a la traducción del artículo original de Richard Holleran, publicado en Microcomputing EE.UU. (Marzo 1984).*

## Saca tu C-64 de su concha

*Añádele nueve comandos del Turtle Graphics al repertorio de tu Commodore 64*

Por Richard HOLLERAN  
(Microcomputing EE.UU.)  
Traducido y adaptado  
por Diego ROMERO

Existen dos aspectos interesantes en mi programa de gráficos: lo que es y cómo lo hace. Este programa añade nueve comandos del Turtle Graphics al Commodore 64, incluyendo Draw, Plot, Move, Left y Right. Cuando ejecutas el programa, coloca las rutinas de lenguaje máquina en la parte superior de la memoria, dejando unos 21 Kb de memoria libre.

Como probablemente sabéis, el Turtle Graphics es una configuración muy popular de gráficos por ordenador en la cual una "tortuga" (turtle) (real o imaginariamente representada) es programada para moverse por la pantalla, dejando un rastro visible tras de sí. En mi versión la tortuga es, desafortunadamente, imaginaria.

### Un vistazo a los comandos

El comando DRAW inicializa la localización y orientación de la tortuga y borra la pantalla. DRAW también fija los colores azul oscuro de fondo y azul claro para el dibujo.

GR#X permite pasar de la pantalla de texto (GR#0) a la de gráficos (GR#1) y viceversa. Esto es útil si quieres volver a ver lo que tenías en la pantalla de gráficos, ya que DRAW borraría la pantalla. Si existe un error mientras estamos en el modo gráfico, el ordenador pasará al modo de texto que nos lo indicará.

COLOR X, Y, permite escoger el color de la pantalla y del dibujo. Por ejemplo, COLOR 6,14 fija dibujos azul claro sobre

# EXCURSION POR LA MICROPROGRAMACION

fondo azul oscuro, que son los ajustes por defecto u omisión. Los números de los colores son los mismos que aparecen en el manual del Commodore 64.

El comando MOVE mueve la

tortuga imaginaria, dejando el rastro visible. Su sintaxis correcta es MOVE X, donde X es un número positivo menor que 511 o una expresión cuyo resultado sea menor que 511. Un valor

mayor que 510 produce una línea de longitud imprevisible. Un valor negativo no afecta a la pantalla.

El comando LEFT X altera la orientación de la tortuga girán-

LINE# LOC CODE

LINE

00001 0000  
00002 0000  
00003 0000  
00004 0000  
00005 0000  
00006 0000  
00007 0000  
00008 0000  
00009 0000  
00010 0000  
00011 0000  
00012 0000  
00013 0000  
00014 0000  
00015 0000  
00016 0000  
00017 0000  
00018 0000  
00019 0000  
00020 0000  
00021 0000  
00022 0000  
00023 0000  
00024 0000  
00025 0000  
00026 0000  
00027 0000  
00028 0000  
00029 0000  
00030 0000  
00031 0000  
00032 0000  
00033 0000  
00034 0000  
00035 0000  
00036 0000  
00037 0000  
00038 0000  
00039 0000  
00040 0000  
00041 0000  
00042 0000  
00043 0000  
00044 0000  
00045 0000  
00046 0000  
00047 0000  
00048 0000  
00049 0000  
00050 0000  
00051 0000  
00052 0000  
00053 0000  
00054 0000  
00055 0000

TURTLE GRAPHICS PARA EL 64  
AUTOR: RICHARD HOLLERAN  
ADAPTADO POR DIEGO ROMERO

DX = \$0000  
DY = \$0005  
XCOR = \$0004  
YCOR = \$000F  
CURANG = \$0014  
STARTX = \$0019  
STARTY = \$001E  
STARTA = \$0023  
DEGRAD = \$0028  
SQUARE = \$002D  
COLOR = \$0061  
FINDDX = \$0085  
FINDDY = \$0084  
PLOT1 = \$00C7  
IGR0 = \$0047  
IGR1 = \$003F  
FLAG = \$7  
TABLE = \$0227  
BITVAL = \$021F  
CHROET = \$0073  
IERROR = \$A43B  
READY = \$A474  
LET = \$A9H5  
EVALX = \$AD9E  
FIXFLO = \$B391  
INPUT2 = \$B7EB  
PLUS = \$B067  
MULT = \$BA23  
MEMFAC = \$BBA2  
FACMEN = \$BBD7  
FLOFIX = \$BC9B  
COSINE = \$E264  
SINE = \$E26B

CAMBIO EN DIRECCION X  
CAMBIO EN DIRECCION Y  
COORDENADA X ACTUAL  
COORDENADA Y ACTUAL  
ORIENTACION ACTUAL  
VALOR X INICIAL (159)  
VALOR Y INICIAL (99)  
ANGULO INICIAL (30PI/2)  
PI/180 PARA CONVERSION  
VALOR PARA 'SQUARE'  
UTILIZADO PARA 'DRAW'  
PARA CALCULAR DX  
PARA CALCULAR DY  
USADO PARA 'MOVE'  
ENTRADA PARA GRAB  
ENTRADA PARA GRAB  
LAPIZ PINTADO PINTA  
TABLA DE PALABRAS  
TABLA DE VALORES DE BITS  
ROUTINA GET DEL SISTEMA  
ROUTINA DE ERROR BASIC  
ESCRIBE 'READY'  
ROUTINA 'LET' DEL BASIC  
EVALUO EXPRESION  
CONVERSION REAL A ENTERO  
ENTRA DOS VALORES  
SUMA EN BASIC  
MULTIPLICACION BASIC  
MEMORIA A ACUMULADOR#1  
FAC#1 A MEMORIA  
ENTERO A REAL  
COSENO EN BASIC  
FUNCION SENO

.MAC XLOYHI  
LDX #C?1  
LDY #?71  
.MND

PREPARA REGISTROS X E Y  
BYTE BAJO DE DIRECCION  
BYTE ALTO

.MAC ALOYHI  
LDA #C?1  
LDY #?71  
.MND

PREPARA REGISTROS A E Y  
BYTE BAJO DE DIRECCION  
BYTE ALTO

.MAC UPDATE  
XLOYHI ?1  
JSR FACMEN  
.MND

PASA FAC#1 A MEMORIA

.MAC FETCH

PASA MEMORIA A FAC#1

## EXCURSION POR LA MICROPROGRAMACION

dola X grados en contra de las agujas del reloj. RIGHT X tiene un efecto similar pero en sentido horario.

PLOT X, Y te permite activar puntos individualmente. El mar-

gen del eje x es de 0 a 319; el del eje y es de 0 a 199. El punto 0,0 está situado en la esquina inferior izquierda de la pantalla. Al igual que en el comando MOVE, los puntos fuera de la pantalla no

son activados, para evitar la posibilidad de hacer un "POKE" en medio del programa que se está ejecutando. Sin embargo en el caso de PLOT, cualquier valor fuera del margen de un byte

LINE#	LOC	CODE	LINE
00055	0000		ALOYHI ?1
00057	0000		JSR MEMFAC
00058	0000		.MND
00059	0000		.MAC ADD ; SUMA MEMORIA A FAC#1
00060	0000		ALOYHI ?1
00061	0000		JSR PLUS
00062	0000		.MND
00063	0000		
00064	0000		.MAC MULPLY ; MULTIPLICA EL FAC#1
00065	0000		ALOYHI ?1 ; POR LA MEMORIA
00066	0000		JSR MULT
00067	0000		.MND
00068	0000		; ?1 LEE EL PRIMER PARAMETRO DE UNA
00069	0000		; LLAMADA A UN MACRO
00070	0000		; ?0 LEE EL BYTE BAJO, ?0? EL ALTO
00071	0000		
00072	0000		* = \$0001
00073	0000		; EL PROGRAMMA INCLUYE UNA LINEA BASIC
00074	0001		BASIC .BYTE \$0C,\$08,\$0A,\$06,\$0E,\$20,\$32
00075	0001	00	
00075	0002	08	
00075	0003	0A	
00075	0004	00	
00075	0005	3E	
00075	0006	20	
00075	0007	32	
00075	0008	30	
00075	0009	36	
00075	000A	32	
00075	000B	00	
00075	000C	00	
00075	000D	00	
00077	000E		; 10 SYS 2062
00078	000E	H9 37	INIT LDA #37
00079	0010	85 01	STA #01 ; HABILITA ROM BASIC
00080	0012	A9 A0	LDA #A0
00081	0014	85 FC	STA #FC ; PUNTERO A ROM
00082	0016	A0 00	LDA #00
00083	0018	84 FB	STA #FB
00084	001A	B1 FB	LOOP1 LDA (\$FB),Y
00085	001C	91 FB	STA (\$FB),Y ; TRANSFIERE EL BASIC
00086	001E	08	INX ; DE ROM A RAM
00087	001F	D0 F9	BNE LOOP1
00088	0021	E6 FC	INC #FC
00089	0023	24 FC	BIT #FC ; CUANDO #FC CONTIENE #3C0
00090	0025	50 F3	BVC LOOP1 ; QUE ES EL FINAL DE BASIC
00091	0027	A9 36	LDA #36 ; ENTONCES NO SALTA.
00092	0029	85 01	STA #01 ; HABILITA EL BASIC EN RAM
00093	002B	H9 81	LDA #81
00094	002D	8D 01 03	STA #01 ; BYTE ALTO DE VECTOR
00095	0030	8D 12 03	LDA #03
00096	0038	8D 06 A6	STA #03 ; DE ERROR
00097	003B	H9 04	LDA #04
00098	003E	8D 00 03	STA #00 ; VECTOR DE ROUTINA USR
00099	003B	H9 D2	LDA #D2 ; NUEVO ANALIZADOR.
			; BYTE BAJO DE ERROR

# EXCURSION POR LA MICROPROGRAMACION

(0-255) causa un error de "illegal quantity".

El comando TAIL UP permite mover la tortuga sin dibujar una línea, probar por ejemplo: TAILUP: MOVE 50. El comando

TAILDOWN baja nuevamente el "lápiz" de la tortuga.

Sería mejor utilizar los comandos originales: PENUP/PENDOWN, pero pendown contiene el comando basic END, y por

tanto no puede ser utilizado desde Basic.

## Encuadrando

Además de estos comandos, puedes utilizar la función USR

LINE# LOC CODE LINE

```
00100 003D 8D 11 03
00101 0040 7F DC
00102 0042 8D 05 H8
00103 0045 85 FE
00104 0047 89 80
00105 0049 85 FE
00106 004B 85 FD
00107 004D 89 08
00108 004F 85 FC
00109 0051 89 71
00110 0053 85 FB
00111 0055 B1 FB
00112 0057 91 FD
00113 0059 C8
00114 005B D0 F9
00115 005D E5 FE
00116 005E E5 FC
00117 0060 A5 FC
00118 0062 C3 06
00119 0064 90 EF
00120 0066 A2 00
00121 0068 A0 5C
00122 006A 18
00123 006B 20 99 FF
00124 006E 4C 02 E4
00125 0071
00126 0071
00127 0071
00128 0071 88
00128 0072 1F
00128 0073 00
00128 0074 00
00128 0075 00
00129 0076 87
00129 0077 46
00129 0078 00
00129 0079 00
00129 007A 00
00130 007B 83
00130 007C 16
00130 007D CB
00130 007E E3
00130 007F F8
00131 0080 7B
00131 0081 0E
00131 0082 FA
00131 0083 35
00131 0084 12
00132 0085 00
00132 0086 40
00132 0087 00
00132 0088 00
00132 0089 00
00133 008A
00134 008A
```

```
STA #0B11 ; BYTE BAJO DE Rutina USR
LDA #10C ; BYTE BAJO DEL ANALIZADOR
STA #10B5
LDA #100 ; PUNTERO DE DESTINO
STA #FE
LDA #119
STA #FD
LDA #SOURCE ; PUNTERO A CODIGO
STA #FC ; FUENTE.
LDA #SOURCE
STA #FB
LDA #FB ; MUEVE EL CODIGO
STA #FD ; AL DESTINO
LDA #FD
END LOOP2
INC #FE
INC #FL
LDA #FL
CMP #10B ; MUEVE TRES PAGINAS
BCD LOOP2 ; EN TOTAL.
LDA #10B ; CARGA X.Y. CON EL NUEVO
LDA #100 ; FINAL DE MEMORIA LIBRE.
LDA #15L ; SALTA A LA RUTINA PERMAN.
JMP #FF99 ; MENOR Y LE DICE LA
JMP #E402 ; MEMORIA LIBRE AL BASIC.

SOURCE
; .BYTE $88,$1F,$00,$00,$00 , 159 EN COMA FLOT.

; .BYTE $87,$46,$00,$00,$00 , 99 EN COMA FLOTANTE

; .BYTE $89,$16,$0B,$E3,$1F ; (3/2)*PI

; .BYTE $7B,$0E,$FA,$35,$12 , PI/180

; .BYTE $80,$40,$00,$00,$00 , .75
```

GR



# EXCURSION POR LA MICROPROGRAMACION

para "encuadrar" (square) un dibujo en la pantalla. Dado el formato de los puntos generados por el C-64, las líneas horizontales y verticales de igual magnitud no son de igual longitud. Prueba

por ejemplo:

DRAW: FOR T=1 TO 4: MOVE  
90: LEFT 90: NEXT

Dependiendo de tu monitor, esto puede parecerse más o

menos a un cuadrado. En mi monitor se parece a un cuadrado cuando la línea en el eje Y es tres cuartos de la magnitud del eje X. Por tanto, el valor por defecto para USR es 0,75.

LINE#	LOC	CODE	LINE
00135	000A		PAR
00136	000A	40	JSR CHRGCT
00137	000B	20 73 00	PLA
00138	000E	00	CMP #0
00139	000F	09 30	BEO GR0
00140	0091	F0 00	CMP #1
00141	0093	09 31	BNE EXIT
00142	0095	D0 17	LDA #43B
00143	0097	A9 3E	LDY #100
00144	0099	A0 00	LDA #47D
00145	009E	A2 7D	BNE SCREEN
00146	009D	D0 06	LDA #41B
00147	009F	A9 1B	LDY #307
00148	00A1	A0 07	LDA #415
00149	00A3	A2 15	STA #D011
00150	00A5	8D 11 D0	STY #D000
00151	00A6	9L 00 D0	STX #D018
00152	00A8	0E 1E D0	EXIT
00153	00A8	00	RTS
00154	00AF		
00155	00AF		COLOR
00156	00AF		
00157	00AF	20 EB B7	JSR INPUT2
00158	00B2	0A	TAX
00159	00B3	0A	ASL A
00160	00B4	0A	ASL A
00161	00B5	0A	ASL A
00162	00B6	0A	ASL A
00163	00B7	05 14	ORA #14
00164	00B9	A2 5C	LDA #45C
00165	00BB	06 FC	STX #FC
00166	00BD	A0 00	LDY #000
00167	00BF	04 FB	STY #FB
00168	00C1	91 FB	STA #4FB,Y
00169	00C3	00	INY
00170	00C4	D0 FB	BNE FILL
00171	00C6	E0 FC	INC #FC
00172	00C8	H6 FC	LDA #FC
00173	00CA	E0 00	ORA #000
00174	00CC	90 F3	BCC FILL
00175	00CE	00	RTS
00176	00CF		
00177	00CF		LEFT
00178	00CF		
00179	00C1	20 9E AD	JSR EVALX
00180	00D2	A0 00	LDA #00
00181	00D4	49 FF	EQW #4FF
00182	00D6	05 00	STA #00
00183	00D8	50 03	BVC RIGHT+3
00184	00DA		
00185	00DA		RIGHT
00186	00DA		
00187	00DA	20 9E AD	JSR EVALX
00188	00DD		MULPLY DEGRAD
00195	00E4		ADD CURANG

# EXCURSION POR LA MICROPROGRAMACION

Cómo funciona

Si el dibujo es demasiado alto, sustituye el valor por uno menor, como 0,65. Si tu dibujo es demasiado ancho, prueba otro valor como 0,8. Una vez que encuentres el valor que necesitas, puedes

sustituírlolos en el programa para que siempre se cargue con el. La sintaxis correcta de la función USR es: A=USR(X), por ejemplo

A=USR(8).

Dada la interesante estructura del Commodore 64, existen al menos tres modos de añadir nuevos comandos al Basic. Está el

LINE#	LOC	CODE	LINE
00202	00E8		UPDATE CURANG
00209	00F2	20 64 E2	JSR COSINE
00210	00F5		UPDATE DX ; DX=COS(CURANG)
00217	00FC		FETCH CURANG
00224	0003	20 6B E2	JSR SINE
00225	0906		MULPLY SQUARE ; DY=SIN(CURANG)*SQUARE
00232	0900		MOVHI DY ; ACTUALIZA DY
00236	0911	40 D7 BB	JMP FACMEN
00237	0914		
00238	0914		PLOT
00239	0914		
00240	0914	20 EB B7	JSR INPUT2 ; ENTRA COORDENADAS
00241	0917	06 02	STX #02 ; ALMACENA YCOR
00242	0919	A9 07	LDA #07 ; YCOR=199-YCOR ESTO PONE
00243	091B	E5 02	SEC #02 ; EL ORIGEN DE LA PANTALLA
00244	091D	85 02	STA #02 ; ABAJO A LA IZQUIERDA
00245	091F	A5 15	LDA #15 ; XCOR EN #14, #15
00246	0921	F0 0B	BEQ OK ; SI XCOR ESTA FUERA DE PANTALLA
00247	0923	C9 01	CMP #01 ; NO LO DIBUJA.
00248	0925	D0 06	BNE NOTOK
00249	0927	A5 14	LDA #14
00250	0929	C9 40	CMP #40
00251	092B	90 01	BEQ OK
00252	092D	60	RTS ; SI YCOR ESTA FUERA DE
00253	092E	A5 02	LDA #02 ; PANTALLA NO LO DIBUJA.
00254	0930	C9 08	CMP #08
00255	0932	B0 F9	BOS NOTOK
00256	0934		
00257	0934		PARA SELECCIONAR EL PUNTO CORRECTO
00258	0934		SE UTILIZAN LAS SIGUIENTES ECUACIONES
00259	0934		CHAR=BASE+40*INT(Y/8)+INT(X/8)
00260	0934		BYTE=YANDT ; BIT=XAND7
00261	0934		AND #07 ; Y GUARDA EL BYTE
00262	0937	A9 00	TAX
00263	0939	85 FC	LDA #00
00264	093B	A5 02	STA #FC
00265	093D	29 FB	LDA #02 ; INT(Y/8)
00266	093F	2A	AND #FB
00267	0940	26 FC	ROL A
00268	0942	2A	ROL #FC
00269	0943	26 FC	ROL A
00270	0945	2A	ROL #FC ; INT(Y/8)*8
00271	0946	26 FC	ROL A
00272	0948	85 FB	STA #FB ; GUARDA VALOR INTERMEDIO
00273	094A	A6 FC	LDA #FC
00274	094C	2A	ROL A
00275	094D	26 FC	ROL #FC
00276	094F	2A	ROL A
00277	0950	26 FC	ROL #FC ; INT(Y/8)*32
00278	0952	85 FB	ADC #FB ; SUMA BYTE BAJO INT(Y/8)*8
00279	0954	85 FB	STA #FB
00280	0956	2A	TAX
00281	0957	85 FC	ADC #FC ; SUMA BYTE ALTO
00282	0959	09 60	UPH #60 ; LE SUMA LA BASE (\$0000)
00283	095B	AA	TAX

# EXCURSION POR LA MICROPROGRAMACION

auténtico wedge o cuña, que desafortunadamente ralentiza la ejecución del intérprete basic (este método podría utilizarse en el VIC-20).

Un modo mejor es copiar la

ROM a la RAM que ocupa las mismas direcciones y entonces modificar los comandos y vectores para satisfacer tus necesidades. El problema es que pierdes un comando de los existentes por

cada uno nuevo. Comencé mi proyecto utilizando este método, pero decidí que no quería eliminar el uso de DEF EN o VERIFY, por citar sólo un par de comandos. De modo que

LINE#	LOC	CODE	LINE
00284	095C	A5 14	LDA #14 ; INT(X/Y)
00285	095E	29 F8	AND #FB ; SUMA BYTE BAJO INT(X/Y)
00286	0960	65 FB	ADC #FB
00287	0962	85 FB	STA #FB
00288	0964	8A	TXA ; SUMA BYTE ALTO
00289	0965	65 15	ADC #15
00290	0967	85 FC	STA #FC
00291	0969	A5 14	LDA #14
00292	096B	29 07	AND #07 ; X GUARDA EL BIT
00293	096D	8A	TXA ; SELECCIONA EL BYTE
00294	096E	B1 FB	LDA (FB),Y ; ACTIVA EL BIT
00295	0970	1D 1F 82	ORA BITVAL,X
00296	0973	91 FB	STA (FB),Y
00297	0975	80	RTS
00298	0976		
00299	0976		DRAW
00300	0976		LDA #E6 ; #E6=AZUL CLARO SOBRE
00301	0976	A9 E6	STA #LARG ; FONDO OSCURO.
00302	0976	85 97	JSR COLOR ; BAJA LAPIZ.
00303	097H	20 01 80	LDA #0F
00304	097D	A0 0F 80	LDA STARTX-1,Y ; FIJA XCOR,YCOR
00305	097F	B9 18 80	STA XCOR-1,Y ; Y EL ANGULO
00306	0982	99 09 80	DEY
00307	0985	68	BNE INITXY
00308	0986	D0 F7	STY #FB ; PUNTERO A PANTALLA
00309	0988	84 FB	LDA #60
00310	098H	A9 60	STA #FC
00311	098C	85 FC	TVA
00312	098E	98	CLEAR STA (FB),Y ; BORRA PANTALLA
00313	098F	91 FB	INY
00314	0991	C8	BNE CLEAR
00315	0992	D0 FB	INC #FC
00316	0994	E6 FC	BPL CLEAR ; PONE CERO EN FAC#1
00317	0996	10 F7	JSR FIXFLO ; INICIALIZA DX Y DY
00318	0998	20 91 B3	JSR FINDDX ; ACTIVA PANTALLA GRAFICA
00319	099B	20 85 80	JSR FINDDY
00320	099E	4C 3F 80	JMP IGR1
00321	09A1		TAIL
00322	09A1		SBC #U ; UN CARACTER DISTINTO DE 'U'
00323	09A1		STA FLAG ; BAJARA EL LAPIZ.
00324	09A3	E9 55	JSR CHGET ; EL PRIMER
00325	09A5	85 97	BNE IGNORE ; CARACTER
00326	09A8	20 73 00	RTS
00327	09AB	D0 FB	
00328	09AD	80	
00329	09AB		MOVE
00330	09AB		JSR EVALEX ; ÉNTRA LONGITUD A MOVER
00331	09AB		LDA #66 ; SI ES NEGATIVA VUELVE
00332	09AB	20 9E AD	BMI DONE
00333	09AE	A5 66	JSR FLOFIX
00334	09B0	20 69	LDA #65
00335	09B2	20 98 BC	LDA #64 ; SI 'MOVE 511', X=#5FF
00336	09B5	A6 65	
00337	09B7	A5 64	
00338	09B9	F0 03	

# EXCURSION POR LA MICROPROGRAMACION

escoji otro método.

Cuando teclas una línea de Basic y pulsas Return, el intérprete basic evalúa la línea y la convierte en "tokens". Si el intérprete espera encontrar una

palabra clave (comando o función) pero no lo encuentra, asume que se trata de un LET, y salta a ejecutar la rutina LET.

Hasta ahora es así. Cambiando este salto de modo que apunte a

una nueva rutina de evaluación te permite añadir nuevas palabras clave aunque tú no las comprimirás (no las convertirás en tokens). Esta rutina no disminuye la velocidad de ejecución de

LINE# LOC CODE

LINE

```

00339 09BE A9 FF          LDA #FF          ; PERO SE INCREMENTA A #80
00340 09BD E8             INX                     ; ES INSOSPECHABLE.
00341 09BE 65 19          STA #19
00342 09C0 86 1A          STX #1A
00343 09C2                NEXT  FETCH XCOR
00350 09C9                ADD DX          ; XCOR=XCOR+DX
00357 09D0                UPDATE XCOR
00364 09D7 20 98 BC      JSR FLOFIX
00365 09DA A6 65          LDX #65
00366 09DC A4 64          LDY #64          ; LO GUARDA PARA PLOT
00367 09DE 84 15          STY #15
00368 09E0 86 14          STX #14
00369 09E2                FETCH YCOR
00376 09E9                ADD DY          ; YCOR=YCOR+DY
00383 09F0                UPDATE YCOR
00390 09F7 20 95 BC      JSR FLOFIX
00391 09FA A6 64          LDX #64          ; ELIMINA YCOR/255
00392 09FC D0 0B          BNE NEXT1
00393 09FE A6 65          LDX #65
00394 0A00 86 02          STX #02
00395 0A02 A5 97          LDA FLAG
00396 0A04 F0 03          BEQ NEXT1          ; SI EL LAPIZ ESTA LEVANTADO
00397 0A06 20 C7 80      JSR PLOT1          ; NO PINTA
00398 0A09 A5 19          BEQ NEXT2
00399 0A0B F0 05          DEC #19
00400 0A0D C6 19          CLC
00401 0A0F 18             STEP  BCC NEXT
00402 0A10 90 B0          NEXT2 LDA #1A
00403 0A12 A5 1A          BEQ DONE
00404 0A14 F0 05          DEC #1A
00405 0A16 C6 1A          CLC
00406 0A18 18             BCC STEP
00407 0A19 90 F5          DONE  RTS
00408 0A1B E0
00409 0A1C                ERROR
00410 0A1C
00411 0A1C                TXA
00412 0A1C 8A             BMI NOERR          ; VUELVE A PANTALLA DE TEXTO
00413 0A1D 30 08          FRR                     ; PARA VISUALIZAR
00414 0A1F 40             JSR IGR0          ; MENSAJE DE ERROR.
00415 0A20 20 47 80      PLA
00416 0A23 66             JMP IERR0
00417 0A24 4C 3B A4      NOERR JMP READY
00418 0A27 4C 74 A4
00419 0A2A                USR
00420 0A2A
00421 0A2A                UPDATE SQUARE
00422 0A2A                JMP FINDDY          ; NUEVO VALOR DE CUADRO
00423 0A31 4C A4 80      ; EL DE 'DY'
00430 0A34                PARSER
00431 0A34
00432 0A34                LDX #800
00433 0A34                LDY #800
00434 0A36 A0 00          SEARCH LDA ($7A),Y
00435 0A38 B1 7A

```

# EXCURSION POR LA MICROPROGRAMACION

las palabras basic normales exceptuando LET. Pero asumiendo que LET "viaja" ya por un tortuoso camino (perdiéndose por 35 palabras), la adición de nuevas palabras es sólo un pequeño desvío.

Si en cualquier momento tu ordenador rechaza los nuevos comandos del Turtle Graphics dándote "Syntax Error", tecllea POKE 1,54 para volver a habilitarlos. Esto no es necesario después de pulsar STOP y RESTORE.

Necesitarás un monitor de lenguaje máquina o un ensamblador para entrar este programa en tu ordenador. El listado 1 está escrito en lenguaje ensamblador; el listado 2 es un volcado hexadecimal de lenguaje máquina. Cuan-

LINE#	LOC	CODE	LINE
00436	0A3A	DD 27 82	CMP TABLE,X ; LO COMPARA CON LAS INICIALES
00437	0A3D	F0 0C	BEQ MATCH ; DE LA TABLA.
00438	0A3F	8A	TXA
00439	0A40	18	CLC ; NO CONCUERDA
00440	0A41	69 08	ADC #50B ; X APUNTA A LA PROXIMA ENTRADA
00441	0A43	AA	TAX
00442	0A44	C9 40	CMP #540 ; ES EL FINAL DE LA TABLA?
00443	0A46	D8 F0	BNE SEARCH ; SI LO ES SALTA A RUTINA LET
00444	0A48	4C A5 A9	JMP LET
00445	0A4E	C8	INX
00446	0A4C	E8	LDA (\$7A),Y ; COMPARA EL RESTO DE
00447	0A4D	B1 7A	CMP TABLE,X ; LA PALABRA CON LA TABLA.
00448	0A4F	DD 27 82	BEQ MATCH
00449	0A52	F0 F7	ORA #500 ; ULTIMO CARACTER EN TABLA?
00450	0A54	09 80	CMP TABLE,X
00451	0A56	DD 27 82	BEQ FOUND
00452	0A59	F0 07	TXA
00453	0A5B	8A	AND #5F8
00454	0A5C	29 F8	LDY #100
00455	0A5E	A0 00	BEQ MISSED ; INTENTARLO OTRA VEZ
00456	0A60	F0 DE	LDA TABLE+1,X ; PONE VECTOR EN STACK
00457	0A62	DD 28 82	PHX
00458	0A65	48	LDA TABLE+2,X
00459	0A66	BD 29 82	PHX
00460	0A69	48	TXA
00461	0A6A	8A	AND #507 ; LONGITUD DE LA PALABRA
00462	0A6B	29 07	TAX
00463	0A6D	AA	JSR CHRGCT ; AVANZA PUNTERO
00464	0A6E	20 73 00	DEX
00465	0A71	CA	BNE MOVEP ; SALTA A LA RUTINA
00466	0A72	D0 FA	JMP CHRGCT
00467	0A74	4C 73 00	; POR MEDIO DE CHRGCT.
00468	0A77		; BITVAL
00469	0A77		
00470	0A77		
00471	0A77		; .BYTE \$80,\$40,\$20,\$10,\$08,\$04,\$02,\$01
00472	0A77	80	
00472	0A78	40	
00472	0A79	20	
00472	0A7A	10	
00472	0A7B	08	
00472	0A7C	04	
00472	0A7D	02	
00472	0A7E	01	
00473	0A7F		; TABLA DE PALABRAS Y VECTORES
00474	0A7F		
00475	0A7F		; .BYTE 'TAI', \$0C,\$81,\$48,\$00,\$00
00476	0A7F	54 41 49	
00476	0A82	CC	
00476	0A83	81	
00476	0A84	48	
00476	0A85	00	
00476	0A86	00	; .BYTE 'RIGH', \$D4,\$80,\$81,\$80
00477	0A87	52 49	
00477	0A8B	D4	

# EXCURSION POR LA MICROPROGRAMACION

do termines de entrar el programa, guárdalo en cinta o disco (antes de probarlo), empezando en la dirección hexadecimal \$0801 (en lugar de \$0800), de este modo

podrás cargarlo y ejecutarlo con LOAD y RUN como un programa en Basic normal.

El programa de demostración (listado 3) te dará algunas ideas

de cómo utilizar los comandos del Turtle Graphics. Observa cómo se dibujan los círculos. Verás, que en muchos casos, cambiando solamente un número

```

LINE# LOC CODE LINE
00477 0A8C 80
00477 0A8D 81
00477 0A8E 00
00478 0A8F 50 4C 4F .BYTE 'PLO', $D4, $80, $B3, $80, $80
00478 0A92 D4
00478 0A93 80
00478 0A94 EE
00478 0A95 00
00478 0A96 00
00479 0A97 40 4F 56 .BYTE 'MOV', $C5, $81, $52, 100, $80
00479 0A9A C5
00479 0A9B 81
00479 0A9C 52
00479 0A9D 00
00479 0A9E 00
00480 0A9F 4C 45 46 .BYTE 'LEF', $D4, $80, $76, $80, $80
00480 0AA0 D4
00480 0AA3 80
00480 0AA4 76
00480 0AA5 00
00480 0AA6 00
00481 0AA7 47 52 .BYTE 'GR', $A3, $80, $31, $80, $80, $80
00481 0AA9 A3
00481 0AAA 80
00481 0AAE 31
00481 0AAC 00
00481 0AAD 00
00481 0AAE 00
00482 0AAE 44 52 41 .BYTE 'DRA', $D7, $81, $1D, $80, $80
00482 0AB2 D7
00482 0AB3 81
00482 0AB4 1D
00482 0AB5 00
00482 0AB6 00
00483 0AB7 43 4F 4C .BYTE 'COL', $D2, $80, $56, $80, $80
00483 0ABA D2
00483 0ABB 80
00483 0ABC 56
00483 0ABD 00
00483 0ABE 00
00484 0ABF .END

```

ERRORS = 00000

## SYMBOL TABLE

SYMBOL	VALUE						
ADD	FFFF	ALOYHI	FFFF	BASIC	0001	BITVAL	821F
CHRGET	0073	CLEAR	09BF	COLOR	0061	COLOR	03AF
COSINE	E264	CURRANG	8014	DEGRAD	8028	DONE	0A1B
DRAW	0976	DX	8000	DY	8005	ERROR	0A1C
EVALEX	AD9E	EXIT	08AE	FACMEN	8BD7	FETCH	FFFF

obte  
dife  
C  
resa  
ción

Como dije al principio, es interesante que te fijas en la utilización de los macros, por ejemplo

cómo se define MULPLY en función de otro marco (ALOYHI) con el parámetro ?1 y luego es utilizado con una macroinstrucción como MULPLY DEGRAD.

En otros ejemplos intentaré equilibrar el VIC para que no se sientan menospreciados sus usuarios.

SYMBOL VALUE									
FILL	08C1	FINDDX	8065	FINDDY	80A4	FIXFLO	B391		B391
FLAG	0097	FLOFIX	809E	FOUND	0A62	GR	088H		088H
GR0	083F	GR1	0097	1ERROR	A43E	IGNORE	09A5		09A5
IGR0	0047	IGR1	803F	INIT	808E	INITXY	097F		097F
INPUT2	87EB	LEFT	80CF			LOOP1	001H		001H
LOOP2	0655	MATCH	0A4E	MEMFAC	8B42	MISSSD	0440		0440
MOVE	09A8	MOVEP	0A6E	MULTPLY	FFFF	MULT	BH28		BH28
NEXT	09C2	NEXT1	0A09	NEAT2	0112	MUERR	0A27		0A27
NOTOK	092D	OK	004E	READY	0A34	PLOT	0914		0914
PLOT1	80C7	PLUS	8067	SHORT	A474	RIGHT	06DA		06DA
SCREEN	08A5	SEARCH	0A30	SHORT	03BE	SINE	E2E6		E2E6
SOURCE	8071	SOURCE	0A2D	STKTH1	8025	STKTHX	8019		8019
STKTY1	801E	STEP	0A10	TABLE	8A27	TAIL	09F1		09F1
UPDATE	FFFF	USR	0A2A	XCOR	800A	YLUYH1			
YCOR	800F								

END OF ASSEMBLY

00990	00	0C	08	0A	00	9E	20	32	
00998	30	36	32	00	00	00	A9	37	
00810	04	01	A9	00	85	FC	0A	00	
00618	14	FB	81	FB	91	FB	C8	08	
00514	7F	E6	FC	24	FC	50	F3	A9	
00428	36	85	01	A9	81	80	01	03	
00330	80	12	03	8D	06	A8	A9	C4	
00236	80	00	03	A9	D2	80	11	03	
00140	FE	DC	8D	05	H8	A9	00	05	
00045	H9	A9	19	85	FD	A9	08	85	
00050	FC	A9	71	85	FB	B1	FB	91	
00058	FD	C8	00	F9	E6	E6	E6	FC	
00060	A5	FC	C9	08	90	E2	00	00	
00068	0A	5C	18	20	99	FF	4C	02	
00070	E4	08	1F	00	00	00	87	66	
00078	00	00	00	83	16	CB	E3	F8	
00080	7B	9E	FA	35	12	80	00	00	
00088	00	00	48	20	73	00	69	C9	
00090	30	F0	0C	C9	31	00	71	A9	
00098	38	0A	C6	A2	7E	00	86	A9	
000A0	18	8C	07	A2	15	8D	11	08	
000A8	00	00	DD	8E	18	00	60	20	
000B0	EB	87	8A	0A	FC	0A	0A	05	
000B8	14	A2	5C	09	FC	00	00	84	
000C0	FC	FB	91	83	C9	0D	FB	E6	FC
000C8	A6	FC	E6	60	98	F3	60	20	
000D0	9E	03	20	95	A6	49	FF	85	6G
000D8	30	03	20	9E	A0	A9	28	A0	
000E0	50	00	20	8B	A9	14	A0	80	
000E8	20	67	B8	A2	14	A0	80	20	
000F0	07	BB	20	64	E2	A2	00	A0	
000F8	60	20	07	BB	A9	14	A0	80	
00100	20	A2	88	20	68	E2	A9	20	
00098	0A	80	20	28	A2	85	A0	80	
00108	80	4C	07	BB	20	68	E7	86	
00118	02	A9	07	F3	E5	02	85	A5	
00200	15	F8	08	C9	01	D8	06	A5	

0928	14	C9	46	90	01	60	85	02
0930	39	C8	80	F9	29	07	A8	A9
0938	00	85	FC	85	FC	2A	F8	2A
0940	26	FC	A6	26	2A	FC	0A	2A
0948	85	FB	A6	26	85	FB	8A	65
0950	26	FC	65	0A	85	14	F8	0A
0958	FC	65	FB	85	8A	65	15	85
0968	FC	A5	14	29	07	AA	B1	F8
0970	10	1F	82	91	FB	60	89	E6
0978	85	97	20	61	80	08	0F	89
0980	18	00	99	09	80	88	00	F7
0988	84	FB	A9	60	85	FC	98	91
0990	FB	C8	D0	FB	E6	FC	10	F7
0998	20	91	83	20	85	80	4C	3F
09A0	80	89	55	85	97	20	73	00
09A8	08	FB	60	20	9E	AD	A5	66
09B0	30	69	20	98	CC	A6	65	85
09B8	64	F0	83	A9	FF	E8	85	19
09C0	86	1A	89	0A	00	80	20	A2
09C8	BB	A9	00	80	30	20	07	85
09D0	A2	0A	0A	80	20	07	85	15
09D8	98	BC	86	A5	A4	64	94	15
09E0	86	14	69	AF	08	80	20	A2
09E8	BB	A9	05	0A	80	30	07	88
09F0	A2	0F	0A	80	20	07	88	20
09F8	98	BC	A6	64	00	08	A6	65
09A0	86	82	85	F7	F0	83	20	C7
09A8	80	85	F5	05	85	C6	19	18
09B0	98	F5	1F	F0	85	C6	85	1A
09B8	18	90	A0	68	4C	3B	A4	4C
09C0	26	A7	80	62	A0	8A	00	D7
09C8	66	4C	A0	8A	A2	00	H0	00
09D0	81	7A	D0	27	82	F0	CC	8A
09D8	18	69	08	CA	A9	40	D0	F0
09E0	4C	A5	89	C8	E8	B1	7A	D0
09E8	27	82	F0	F7	09	80	D0	27
09F0	82	F0	87	8A	29	F8	80	60

## EXCURSION POR LA MICROPROGRAMACION

```

:0A60 F0 DE 80 28 82 48 80 29
:0A68 82 48 8A 29 07 AA 20 73
:0A70 00 CA D0 FA 4C 73 00 80
:0A78 40 20 10 00 04 02 01 54
:0A88 41 49 CC 81 48 00 00 52
:0A88 49 47 48 D4 80 81 00 50
:0A90 4C 4F D4 80 88 00 00 48
:0A98 4F 56 C5 81 52 00 00 4C
:0AA0 45 46 D4 80 76 00 00 47
:0AA8 52 A3 80 31 00 00 00 44
:0AB0 52 41 07 81 10 00 00 43
:0AB8 4F 4C D2 80 56 00 00 DF
    
```

### LISTADO 3

```

1 REM DEMOSTRACION DEL TURTLE
10 DRAW:COLR14,6:TAILUP:RIGHT150:MOV
E150:LEFT150:TAILODOWN
20 FORG=1TO36:IFORT=1TO2:FORR=1TO8:MO
VE9:LEFT10:NEXT
    
```

```

30 FORR=1TO6:MOVE9:RIGHT10:NEXT:RT:R
IGHT170:NEXT:GOSUB300
40 FORT=4TO8STEP2:R=360/T:DRAW
50 FORR=1TOT:FORG=1TOT:MOVE9:LEFTR:N
EXT:RIGHTA:NEXT
60 GOSUB300:NEXT
70 DRAW:POKE53280,0:COLR0,2:R=160:IF0
RT=1TO200:MOVET:LEFTR:NEXT:GOSUB300
80 FORT=13TO15STEP2:R=720/T:DRAW
90 FORR=1TOT:FORG=1TOT:TAILODOWN:IFG/
2=INT(G/2):THENTAILUP
100 MOVE9:LEFTR:NEXT:RIGHTA:NEXT
110 GOSUB300:NEXT
120 DRAW:POKE53280,0:COLR0,6:FORT=1T
O24:FORR=1TO36
130 MOVE10:LEFT10:NEXT:RIGHT15:NEXT:
GOSUB300
140 DRAW:COLR6,8:R=89.5:FORT=1TO2385
TEP1.5:MOVET:LEFTR:NEXT:GOSUB300
299 GR0:END
300 FORV=1TO1000:NEXT:POKE53280,14:R
ETURN
    
```

## COMENTARIOS MODORE WORLD

(Viene de pág. 51)

La cuarta sección, titulada "Más sobre programación", da más detalles sobre la utilización de técnicas de programación avanzadas, ya que la segunda sección era muy elemental y no detallaba todas las posibilidades del Commodore 64. En esta sección del libro encontramos capítulos dedicados a instrucciones condicionales (IF/THEN), cómo utilizar las sentencias DATA/READ, cómo poder adaptar los programas que deseamos a nuestras necesidades particulares, y una serie

de programas algo más complejos que los que habíamos encontrado hasta este momento.

Las cuatro secciones anteriores constan de un total de dieciocho capítulos, que cumplen perfectamente su función: servir de guía para el usuario durante las primeras semanas o meses de utilización del ordenador, y no decirlo todo sobre el, ya que esto último supondría utilizar un manual de más de mil páginas y aún quedarían puntos sin aclarar. La finalidad del libro la deja bien clara el autor en la introducción, en la que dice que tiene un alcance limitado. Creemos que aunque esté limitado, mucho más lo están los manuales y dejan demasiados puntos sin aclarar, por lo que esta obra es muy útil para todos los principiantes.

Para terminar el estudio, encontramos cincuenta y tres páginas destinadas a los apéndices. Los temas tratados en esta parte del libro podrían confundir el estudio inicial del ordenador, por lo que pueden verse al reparar posteriormente lo aprendido. Describen todas las peculiaridades y complejidades del ordenador personal C-64, y son: El comando Poke, la instrucción Peek, los sprites, los sonidos, qué hacer con sus propios programas, una descripción de todas las palabras clave del lenguaje basic, y un apéndice dedicado a las tablas de códigos de pantalla, ASCII, mapas de memoria de pantalla y color, valores para obtener las notas musicales, mapas de los registros de sprites y bases del control de sonido del Commodore 64.

## TIRADA DE COMMODORE WORLD

### Certificación de 28.000 ejemplares

Según certificado firmado por la auditoría internacional DELOITTE HASKINS SELLS, S.A. el 5 de marzo de 1984, la tirada de Commodore World es de 28.000 ejemplares. Este certificado se encuentra en las oficinas de S.I.M.S.A. para comprobación de cualquier persona o entidad que lo desee.

### Oferta de trabajo

## PROFESOR JEFE

Enseñanza de Informática (Barcelona)

- Para responsabilizarse de los cursos de informática en una entidad con experiencia en este campo. Contará con 140 alumnos y coordinará el trabajo de cuatro colaboradores.
- Trabaja con ordenadores Commodore. Debe tener dominio del lenguaje Basic y conocimientos de Logo. Se valorarán también otros lenguajes.
- La edad de los alumnos es de 10 a 18 años. Ocasionalmente impartirá cursos de introducción a la informática para profesionales y desarrollará proyectos informáticos.
- Debe poseer título universitario (Facultad o Escuela Técnica). Se valorará experiencia en la docencia.
- La dedicación es de un mínimo de 3,5 horas diarias, por las tardes. Se incorporará a su trabajo en enero de 1985.

Interesados, concertar entrevista escribiendo a la dirección:

FP

Casanovas, 164-166 — 1º 1ª

08036 BARCELONA

(indicando nombre y teléfonos de localización)



# MARKET CLUB

Servicio gratuito para nuestros lectores **PARTICULARES**. Los anuncios serán publicados durante 1 mes. Debido a ciertos abusos que nos han sido comunicados, los anuncios gratuitos de Market Club **SOLAMENTE** serán publicados si vienen con nombre, apellidos y dirección completa.

## MERCADILLO

- Compro Controlador C8 de 8 teles (precio a convenir) o cambio por cartucho SUPEREXPANDER +3K para el VIC-20 y un cassette con juegos y programas también para el VIC-20. Vendo libro "INTRODUCCION AL LENGUAJE DE PROGRAMACION BASIC" para el VIC por 1.000 ptas., cartucho SUPEREXPANDER +3K por 7.000 ptas. (todo por 9.000 ptas.) (VIC-20). Me interesaría ponerme en contacto con alguna persona que tuviera todos los números de CLUB COMMODORE o con alguien que tuviera esquemas de algún modém telefónico o de algún RTU-CW para CBM 64. Llamad de 19.30 a 23.05 h. a José de la Fuente, al Tel. (93) 242 51 70. 08183 Barcelona. (Ref. M-174).
- Desearía comprar instrucciones sobre el manejo y programación del Commodore 64 en castellano. Carlos Iglesias, C/La Llacuna, 4. Andorra la Vella. (Principado de Andorra). (Ref. M-175).
- Hoya soy un VIC-cioso de Barcelona y buscaría alguien que me vendiera el Super Expander del VIC-20. Precio a convenir. Llamad al teléfono 794 07 09. O escribir al Paseo de la Misericordia nº 18 de Canet de Mar (Barcelona). Si llamáis preguntad por Luis Maria. Llamad preferiblemente festivos o laborables de 19 a 23 h. (Ref. M-176).
- Desearía comprar VIC-20 y Commodore 64. 1. Impresora tracción y otra tracción y fricción adyacente a los dos ordenadores. CARLES BONJOCH / C/Mosen Annin nº 1. Pral. Llerda (973) 24 21 08 (Ref. M-177).
- El Sr. Jiménez busca una fotocopia del Nº 1 agotado. Quien pueda suministrarla se ponga en contacto con él: Angel Jiménez Camino, Avda. de Barcelona, 12-2º I. Tels. 25-5643. Córdoba. (Ref. M-178).
- Vendo para VIC-20 cassette CN2 - 9.000 ptas., memoria de 8K a 16K Ram por 11.000 ptas. cartucho de superexpander con 3K de memoria Ram 7.000 ptas. cartucho de monitor de código máquina 4.000 ptas. Jesús Prieto. Calle Barranón, 25. 28011 Madrid. Tel.: 463 72 30. (Ref. M-179).
- Busco programas sobre VIC-20 + 64K Ram sobre arquitectura, cálculo estructuras, pórticos, etc. Gratifico información. Máximo Martín. C/Orihama, 75-47ª izda. Tel. 204 460, 15002 La Coruña (Ref. M-180).
- Vendo para VIC-20 cartucho de juegos: VIC-AVENGER referencia (1901) por 4.000 ptas. o lo cambiaria por el libro "Acceso rápido al VIC-20 (Getting acquainted with your VIC-20)". Ofertas llamar al (958) 272217, preguntar por Julián (hijo) Manglaro García, C/Acría de Canasteros, 16, 4º A. 18012 Granada. Llamad de 10 a 11 de la noche (Ref. M-181).
- Vendo VIC-20 nuevo por 24.000 ptas., regalo 2 libros de Basic curso de introducción al Basic en cinta (1.º parte), y unos juegos. Intercambio programas en disco o cassette para C-64. Interesados dirigirse a: Blas Sánchez Navarro, C/Cruzadas, 58, Sabadell. Barcelona. Tel. 711 00 49 (Ref. M-182).
- Intercambio programas para el C-64. Francisco Verdú, C/Barcelona 65, 5.º. 17001 Girona. Tel. (972) 210 663 (Ref. M-183).
- Vendo VIC-20 en perfecto estado (12-83) por 25.000 pesetas más cassette C2N (10.000 ptas.) más guía del Usuario del VIC (1.000 ptas) más dos cursos Introducción al Basic (1.500 ptas.) todo junto por 39.000 ptas., también por separado. Regalo libro "Acceso rápido al VIC-20". Manuel Aranda Atienza, C/Río Sella, 10 - 4º B. Mostoles - (Madrid). Tel.: 617 97 31. (Ref. M-184).
- Pongo un C-64 con el cassette C2N y deseo intercambiar programas de utilidad en general, y programas relacionados con la radioafición. Interesados dirigirse al P.O. Box 1, Peñaflor (Sevilla) o llamar de 21 a 23 horas, de lunes a viernes al 807034 Antonio Cano Quesada, C/Peñaflor, 34, Peñaflor (Sevilla). (Ref. M-185).
- Cambio programa MASTER-64 por cartucho lenguaje Forth o vendo por 18.000 ptas. que también permite correr programas de las series 4.000 y 8.000 en el C-64. Fernando Franco, C/R Felipe de Paz, 12, 15º I-1º 08028 Barcelona. Tel.: 339 87 95. (Ref. M-186).
- Vendo Commodore 64 con cassette, cartucho de Fútbol y Joystick, nuevo con garantía a precio interesante. Llamad a: Magin Borrell, San Carlos, 23. Badalona. Barcelona. Tel. (93) 389 43 08. (Ref. M-187).
- Vendo C-64 con cassette 65.000 ptas. José García González, C/Candoso, 2-2º F. Torrijón de Ardoz. Madrid. Tel. (91) 676 84 09. (Ref. M-188).
- Intercambio programas del C-64. Vendo curso de introducción al C-64. 1 (1.500 ptas.) José Cortes A. C/Aurora, 19-4º I-1.º. 08001 Barcelona. Tel. 3296631. (Ref. M-192).
- Desearía cambiar programas de Juegos u otras aplicaciones para C-64. Interesados llamar Tel. (93) 432 16 87 o escribir a Eduardo Melero Vázquez / C/Mármol, 3, 3º 2.º. 08004 Barcelona (Ref. M-190).
- Vendo ordenador VIC-20 con ampliación universal de 64K, 32 librerías y 32 ocultas, datasette, joystick y tres juegos, todo por 55.000 ptas. Vendo ampliación de 16 K por 11.500 ptas. Andres Lacur Limiñana, C/Urgel, 158, 4º piso. 08036 Barcelona. (Ref. M-191).
- Vendo calculadora CASIO PB300 programable en Basic. Sin estrenar. Llamad por las tardes al teléfono (929) 891106 o escribir a Isabel Rodríguez Lara. Plaza 3 de Noviembre, nº 1-6º C. Torrelavega (Cantabria). (Ref. M-193).
- Cambio cualquier tipo de programas y experiencia para el C-64. Sergi Arbós Mauri / C/Forn del Vidre 16, 3º 4º. Vilanova i la Geltrú (Barcelona). Tel. 893 18 31. (Ref. M-193).
- Vendo C-64, además regalo, el datasette de Commodore, algunas revistas, y el manual en castellano de la Commodore. **Todo completamente a estrenar por ti.** Precio 70.000 (Valor en tienda 92.000) Negociables en cierta manera. Tel. 889 15 63. (Hora de comidas). Angel Antonio Corral Gutiérrez, C/Parque Falco, nº 1. Alcalá de Henares (Madrid). (Ref. M-194).
- Vendo VIC-20 con datasette por 29.900 ptas. Todo funciona de maravilla. El VIC-20 es de octubre del 83 y el datasette de enero del 84. Llamad al 357 04 29 de Barcelona. Preguntar por Germán Costa Aixas, C/Agudells, 56, 08032 Barcelona. (Ref. M-195).
- Intercambio juegos para el COMMODORE 64, todos en cinta, también tengo programas de aplicaciones, los que se interese llamen al número 229 07 42 y pregunten por Francesc. Si les hace más ilusión escribirme, escriban a: Francesc Guasch Ortiz, Calle Jerez, 40 Torre, 08032 Barcelona. (Ref. M-196).
- Vendo revista de Club Commodore nº 3 y 4, por 200 ptas. cada una más gastos de envío (no son fotocopias). Tel. (972) 21 82 24, preguntar por Victor Roca, C/Maluquer Salvador, 24, 3º 3º. 17002 Girona. (Ref. M-197).
- Cambio, vendo, o intercambio material VIC-20, cassette 3K+ superexpander, ampliación de 16K, cartucho, sargón II, Chess, más máquina fotográfica reflex para estremo, por Commodore 64, al mejor postor. (Ramón Romero García Tel. 891 25 90, C/Verde Flamingo, 1. 1.º izda. Aranjuez. (Ref. M-198).
- Vendo C-64 con cassette commodore comprado en Febrero del 84 suscripción a la revista desde el número 1 y muchos programas por 75.000 ptas. Francisco Rodríguez González, 222 380 o 255 635 de Badajoz. Avd. Juan Sebastián Elcano, 33, 6º B (Ref. M-199).
- Vendo VIC-20, cassette C2N, guía de referencia del programador y revistas Commodore". José Antonio Serrano Bergali, C/Avda. de Kansas City, 32, Tel. (954) 57 33 08, 41007 Sevilla. (Ref. M-200).
- Se vende por cambio de ordenador 1 ordenador VIC-20; 1 ampliación de memoria de 3K, 8K, 16K, siendo opcional conectar los 3, 8 o 16K; 2 video juegos en cartucho; 3 cintas de juegos; 1 programa de utilidad para el VIC-20; 1 cinta de columnas; 1 cinta con juegos; 1 guía de referencia del programador. Los dos cursos de introducción al basic, más las cintas.

- José M. Framit Campaña, C/Concepción Arrenal, nº 1, 4º Izda. Telf.: 958 - 28 50 04, Granada 18012 (Ref. M-201).
- Vendo VIC-20, con guía de referencia, lenguaje de programación Basic, los dos cintas y un montón de juegos, todo por 22.000 pesetas. Llamad al 333 09 41, José Martín Torres, C/Desamparados nº 22, 3º piso. Barcelona. (Ref. M-202).
- Vendo VIC-20 (diciembre 83), más datasette, más Joystick (model 1311), por 35.000 ptas. Regalo muchos programas y juegos y 10 cintas vírgenes de 60 min. y mucha información y revistas. Daniel Bassas Pablo, Pl. Marqués de Camps, nº 16, 3º 2.º. 17001 Girona. Tel. (972) 212 726. (Ref. M-203).
- Si alguien está interesado en obtener a bajo precio, alguno de los volúmenes de "INTRODUCCION AL BASIC" (I y II) o el "Manual de Referencia del programador", le ruego se ponga en contacto conmigo lo antes posible. Interesados ponerse en contacto conmigo: Patricio Blanco C/Can Noguera, nº 52, 2º B. La Garriga (Barcelona). (Ref. M-204).
- Vendo Commodore 64, Unidad de cassette, introducción al basic del VIC-20 la unidad de disco VC-1541. Todo está en perfecto estado, ya que cambio de ordenador, lo vendería todo por 100.000 pesetas. César Pascual, Estación Renfe, s/n. Manresa. Tel. 872 26 50. (Ref. M-205).

## CLUBS

- Club de amigos. Foto Estudio 2. Plaza de Sombrereros - 2. Palma de Mallorca. Tel.: 251 31 62. (Ref. C-1).
- Club de programación Alaiá. Tels.: 254 480-257 704. Pamplona. (Ref. C-4).
- Club de Commodore de Albacete. Fernando Martínez Guerrero. La Roda, 39, 5º D. Albacete. (Ref. C-5).
- Círculo de Duranens Vigues. Vigo, C/Venezuela, 48. Entrechans. Tels.: (986) 410 683-422 519. (Ref. C-6).
- Asociación Manchega de Usuarios de CBM-64 (A.M.U. C-64), C/Blasco de Garay, 10. Albacete (Ref. C-13).
- Club MICRO-ESPLAI. Gran Via, 682. Pral. (Entre Bruch y Geronla) de Barcelona. (Ref. C-14).
- Interesados en la formación de un Club de informática en Granada capital, dirijanse por correo a C/San Jerónimo, 23-2º, 18001 Granada. Miguel A. Puerta Carrasco. (Ref. C-20).
- Si queréis que hagamos un club "Vic-20" poneros en contacto con Antonio Recober. Telf.: 32 20 46 Málaga. Si algún 64 quiere hacer el camino a nuestro lado, podemos intentarlo. (Ref. C-29).
- Desearía contactar con interesados en formar un club de usuarios del C-64 en Granada. Antonio Toral Peinado, C/Puente nº 3. Illora (Granada). Tel. 958/46 30 09. (Ref. M-30).
- Para formación club nacional de CBM Series 2000 y 3000 exclusivamente, ponerse en contacto por carta con Artemio González Pérez. Fernando Arocena, C/Alfonso X el Sabio, 32-42 44, 38009 Santa Cruz de Tenerife. (Ref. C-31).
- Me gustaría contactar con alguien de Morla para compartir experiencias y/o formar un club de C-64 (y VIC-20 si hace falta). Ya tengo local. Mi teléfono: 954 319 314. (Ref. C-32).
- Ha nacido el club "Brian & Chip" para usuarios del C-64 y Spectrum 16K. Crearemos una revista propia. Intercambiamos programas listados o en cinta, escribiendo desde cualquier rincón de España. Francisco Reig López, C/Marina, 261, 3º 5º, 08025 Barcelona. (Ref. C-33).
- Agradezco la dirección o teléfono de algún socio de mi ciudad o alrededores, a fin de ponerme en contacto con él para ver la posibilidad de formar un club de usuarios. José Esteban Lluar Melis, Daotz, 2-1º 2º P.º. Sagunto (Valencia). (Ref. C-34).
- Grupo de usuarios de Commodore en Tarragona.

gona, estamos intentando formar un club lo más sólido posible, no nos mueve ningún interés lucrativo. Alfred Blasi. Tel: (977) 547 307. Tarragona. (Ref. C-15).

• Deseo contactar con niños/as que deseen formar un club de ordenadores Commodore 64 y VIC-20 C/Avda. del Norte nº 8 atc. 3.º Hospitalet (Barcelona). Manuel González Martínez. Tel: 93-240 23 50. (Ref. C-16)

• Estamos montando una actividad de informática en el Club Los Chupos para chavales, sin ningún fin lucrativo. Javier Huguet, C/Antonio Menchaca, 19. Teléf. 469 34 61. Neguri. (Ref. C-18).

• Me gustaría contactar con usuarios de la zona de Mataró. Dirigirse a: Marcel Planagumá. Pge. Mir 1 Borrell, 1-1º 2.º Mataró (Barcelona). Tel: (93) 7983127 (Ref. C-42).

• Desearía información sobre clubs en existencia y gente que quiera formar uno en Pontevedra de Commodore 64, Carlos Parada Gandos. C/Padre Fernando Olmedo, 4-4º B. 36002 Pontevedra. Tel: (986) 85 69 21. (Ref. C-43).

• Ha nacido el Club de Informática y videojuegos, el que no escriba recibirá su carnet, con nº, intentaremos crear un presidente por ciudad (el 1º en

escribir de cada una). Juan A. Manso. C/Illescas, 81. 28024 Madrid (Ref. C-44)

• Zaragoza. Club de amigos del VIC-20 y C-64 en ESPUMOSOS los sábados de 9.30 a 12.30. Daniel Fdez. de Velasco, C/Gran Vía, 39, Zaragoza. (Ref. C-45).

• Les ruego si puede ser que me pongan en contacto con algún usuario del VIC-20. Daniel Roig. C/Castilla, 27. Ibiza (Balears) (Ref. C-46).

• Me gustaría contactar con alguien de Figueras o alrededores para intercambiar programas o formar un Club Commodore 64. Josep Teixidor Planas, Vilasacra, 2. El Far D. Empordà (Girona). Tel: 508649. (Ref. C-47).

• VIC-20 Desearía un intercambio de experiencias en cuanto a programación, aplicaciones al campo de la radiofrecuencia (interfaces) y programas diversos. ECG M.º Lorenzo Sabater Coll. C/Aguá, 2. Esporlas, Mallorca. Balears. (Ref. C-48).

• Estamos formando un club de usuarios del COMMODORE 64 en Las Palmas. Desearía contactar con chicos y chicas para intercambio de programas y experiencias. Interesados dirigirse a: César Montenegro Armas, C/Pío XII, 56 2-D. Tel: 24 60 61. (Ref. C-49).

• Desearía ponerme en contacto con usuarios del VIC-20 para intercambiar programas e información. Carles Balahà, C/Tivoli, nº 11, 5º A. Reus (Tarragona). (Ref. C-50).

• Som dos nois de SABADELL que estem interessats en formar un club d'usuaris d'ordinadors COMMODORE i SINCLAIR. Els interessats o interessades, poden escriure a: Daniel Ribes Garrotera, C/Via Aurelia nº 24, 2º 3º Sabadell. També demanarà si es possible informació sobre clubs ja existents. (Ref. C-51).

• Desearía contactar por correo con usuarios del VIC-20, tanto usuarios nacionales como extranjeros. Escribir a Mº Mercedes de la C/Marqués del Munt, 25 (Guía de Gran Canaria). (Ref. C-52).

• Se ha creado "COESPECTRUM 16": un nuevo Club, nacido para todos los usuarios de CBM 64, Spectrum 16k y Dragon 32 y 64. Pero para que todo ello se haga realidad, necesitamos colaboradores y muchas ideas. Se intercambiarán toda clase de programas o listados. Se creará una revista. Escribir a Avd. de la Aurora, 57, 12º A. (29006) Málaga, desde cualquier punto de España, se contestará o llamará a: Angel, Tel: 321799, o Emilio, Tel: 398169. (Ref. C-53).

## "Commodore World" aparte de venderse generalmente en kioscos, se encuentra asimismo a la venta en las siguientes distribuciones de Commodore y librerías.

### BILBAO

• Bilbomiro, S.A. c/Aureliano Valle, 7. 48010 Bilbao. Teléfono: (94) 443 43 51.

### BURGOS

• E.L.S.A. c/Madrid, 4. Teléfono: (947) 20 46 24.

### CACERES

• Informática Vivas White. San Pedro, 8. Teléfono: (927) 24 40 96.

### CADIZ

• Video Computer. Comandante Gómez Ortega, 59. Algeciras. Tel: (956) 65 39 02

### CASTELLON

• Horizontes, c/Navarra, 76.

### CIUDAD REAL

• Electrónica Turrillo. c/Pedrería Baja, 7. Teléfono: (926) 22 38 67.

### LA CORUÑA

• Cetronic, S.L. c/Palomar, 22 Bajo. Teléfono: (981) 27 26 54

• Gesty Computer. Avda. Romero Donallo, 25. Tel: (981) 59 87 54. Santiago de Compostela

• Sanlucia, S.L. San Luis, 46 al 50. Teléfono: (981) 23 07 49.

### GERONA

• Digit Informática, c/Avda. 11 Setembre, 7. Olot. Teléfono: (972) 26 94 01.

• Microchip. c/Aigua, 3. Olot. Tel: 26 36 63.

• Regiscompte, S.A. c/Emilio Grahit, 17 Bis. Teléfono: (972) 21 99 88

### GRANADA

• Rafael Moreno Torres. Recogidas, 24. Edif. Castro. Teléfono: (958) 26 20 50

### HUELVA

• Computlog, S.A. c/Tendaleras, 15. Teléfono: (955) 25 81 99

### IBIZA

• Kelson. Plaza España, S/N. San Antonio Abad. Teléfono: (971) 34 13 09.

### LEON

• Logdata. c/Burgo Nuevo, 4. Teléfono: (987) 20 42 89

• MicroBierzo. c/Carlos I, 2. Teléfono: (987) 41 74 21 (Ponferrada).

### MADRID

• Electrónica Lago. c/Barquillo, 40. Key Informática. c/Embajadores, 90. Teléfono: (91) 227 09 80.

• Librería García Peña, c/Cavanilles, 52.

• Micromundo, S.A. El Zoco. Majadahonda. Teléfono: (91) 638 13 89.

• Micros Garden, c/Francisco Silveira, 19. Teléfono: (91) 401 07 27

• País de los Microordenadores. Bravo Murillo, 18. Tels: (91) 446 33 17 - 446 37 12.

• Remshop. Galileo, 4. Teléfono: (91) 445 28 08

### MALAGA

• Informática Martínez, S.A. c/Cristo de la Epidemia, 90. Teléfono: (952) 26 37 68.

### MURCIA

• Procoinsa. Ronda Norte, 27. Teléfono: (968) 23 94 49.

### ORENSE

• Bermello. Gral. Franco, 123.

### PAMPLONA

• Itar Computers, S.A. c/Alfonso el Batallador, 16 (Trasera). Tel: (948) 27 64 04.

### SALAMANCA

• Info-Gest c/Ronda del Corpus, 2-1º centro. Teléfono: (923) 21 59 93.

### SAN SEBASTIAN

• Donmircio, S.A. c/Arrasate, 6. Teléfono: (943) 42 35 10.

### SANTANDER

• Librería Hernández. San Francisco, 15. Teléfono: (942) 22 53 30.

### SEVILLA

• Papelería Mora. c/Santa Cruz, 5. Ecija. Teléfono: (954) 83 14 80.

### TARRAGONA

• Comercial Informática de Tarragona, S.A. c/Gasómetro, 20. Teléfono: (977) 23 08 53.

• Oficomplet. Plaza de la Cinta, 6. Tortosa. Teléfono: (977) 44 14 50.

### VALENCIA

• Inlekca. c/Remedio, 38-bajo-dcha. Sagunto. Teléfono: (96) 246 21 10 - 246 09 29.

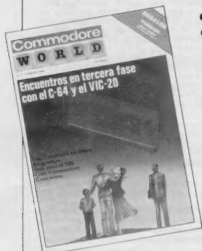
• Librería Mayte. Pintor Benedito, 3. Teléfono: (96) 325 28 83

### VALLADOLID

• Chips + Tips, S.A. c/Juan de Juni, 3. Teléfono: (983) 33 40 00.

### ZARAGOZA

• ADA Computer. Centro Independencia. Pº Independencia, 24-26. Tel: (976) 29 85 62.



### ALICANTE

• Casa Wagner, c/Juan Carlos I, 37. Elda. Teléfono: (965) 39 03 96

• Ferrería Progreso, c/General Jordana, 28. Elda. Teléfono: (965) 38 11 45.

### BADAJOS

• Control Sistemas. Avda. Santa Marina, 25A. Teléfono: (924) 25 88 00.

### BARCELONA

• Catina Informática, c/de L'Esclasia, 15. Teléfono: (93) 784 27 17. Terrasa

• Comercial Claper, c/Mariano Maspons, 4. Granollers. Teléfono: (93) 870 45 42.

• Computer Service. Avda. Abad Marcet, 325. Tarrasa. Teléfono: (93) 788 63 77.

• Computerhard c/s. Jaime, 48. Granollers. Teléfono: (93) 870 09 19.

• Electrodomésticos Mirambell, c/Rabal, 45. San Sadurn d'Anoia. Teléf.: (93) 891 11 34

• Gadesa. Les Valls, 12-14. Sabadell. Teléfono: (93) 725 25 43

• Librería Emilia Pérez Radua, c/Mayor, 35. Castell de Valles. Teléfono: (93) 714 89 51.

• Librería Michel. Rda. Guinard, 1. Sarrià de la Noya. Tel: 145. Mollet. Teléfono: (93) 593 75 01.

• Radio Watt. Paseo de Gracia, 30. Teléfono: (93) 237 11 82.

• Sonimovil. c/Alcalde Armengou, 53. Manresa. Teléfono: (93) 873 78 17.

• Ministradora Z, S.A. Av. Barbera, 49-51 Sabadell. Teléfono: (93) 710 56 66.

• Tronic. Bigay, 11-13. Tel: (93) 212 85 96.

**brother**

BROTHER, entre los cinco mayores productores del mundo de máquinas de escribir e impresoras para la informática fue seleccionada como la marca oficial de los Juegos Olímpicos de Los Angeles 84. Con una antigüedad de 50 años como fabricante hoy día BROTHER cuenta en servicio con más de 12 millones de máquinas de escribir las cuales tienen hasta 80 tipos de diseños y teclados para 22 diferentes idiomas. Sus aproximadamente 20.000 empleados y 7 plantas mantienen una producción corriente anual de 1 millón de aparatos, junto con otros productos BROTHER.



## Impresoras de margarita

### HR-1:

Muy duradero y fiable, este modelo tiene un carro ancho que permite imprimir 132 caracteres en un papel de 420 mm. de ancho, ideal para grandes trabajos de oficina. Sus cassettes de cinta de imprimir son excepcionalmente económicos y permiten ahorrar hasta 7 ptas. en un folio de texto de 1.000 caracteres sobre otras impresoras de características similares. Se puede acoplar un alimentador de tractor para papel continuo o un alimentador automático de hojas sueltas.

### HR-15, HR-25, HR-35:

Una familia de impresoras de alta tecnología y resolución en su sistema de impresión, tipo margarita, nos ofrecen una compatibilidad con una amplia selección de hardware existente en el mercado, en adición a la amplia selección de procesadores de datos y software. Todos los modelos tienen la cualidad de imprimir en dos colores (negro y rojo), posibilidad de remarcar textos, impresión de super y sub-índices y control de impacto para copias múltiples. Entre otras opciones se encuentran un alimentador automático de papel cortado con un máximo de 200 hojas sin parar y un alimentador de tractor que permite utilizar papel continuo. El HR-15 incorpora una memoria de tampón de 3kbytes (5 opcional) con la posibilidad de conectar un teclado para convertir el conjunto en una máquina de escribir electrónica. Imprime a una velocidad de 13 caracteres/seg. Los otros dos modelos nos ofrecen alta velocidad de impresión, 23 caracteres/seg. para el HR-25 y 32 caracteres/seg. para el HR-35. Este último modelo posee 7kbytes de memoria de tampón y ofrece 25 kbytes como opcional. Con estas memorias el ordenador al que está conectado la impresora se encuentra libre para realizar otras operaciones prácticamente en el momento en que se da el comando de "IMPRIMIR".



## Impresoras de matriz

### HR-5:

Imagine Ud. una impresora de que imprime gráficas y diagramas unidireccionalmente y textos bidireccionalmente en papel corriente, y que además cabe en su cartera dejando el suficiente espacio para acomodar su ordenador portátil u ordenador personal. Este ordenador es más que eso! Ud. puede imprimir cuando y donde Ud. quiera puesto que puede operar con 4 pilas de linternas convencionales. Puede imprimir una página entera de 80 columnas de su texto sin apenas sonido. Incorpora el interface de CENTRONICS o el RS-232. Su cabeza impresora de alta calidad puede imprimir en papel normal o térmico a una velocidad de 30 caracteres/seg.

### M-1009:

La impresora de matriz BROTHER M-1009 ha sido diseñada teniendo en mente la demanda tan grande que existe en fabricar una impresora de bajo costo y alta calidad accesible al bolsillo de aquellos que disfrutan de ordenadores personales en sus hogares.

Entre sus características se encuentran una cabeza impresora de matriz con una garantía de 20 millones de caracteres de vida, imprime a 50 caracteres/seg., bidireccionalmente para los caracteres corrientes y unidireccionalmente para gráficas, diagramas, y super y sub-índices. Baja en sonido, la M-1009 posee además otras funciones de alto precio y calidad como el 96 ASCII II con caracteres gráficos e internacionales, interface de Centronics o RS-232 C. La M-1009 imprime en papel cortado, con alimentadores de hojas incorporados, y con papel continuo. El M-1009 es una unidad eficaz que rendirá largos años de servicio a un costo mínimo de mantenimiento.



### 2024L:

El extraordinario modelo, BROTHER 2024L - ¡Dos impresoras en una!. A 80 caracteres/seg., imprime con calidad impecable para tratamiento de textos, y apretando un botón, imprime a la alta velocidad de 160 caracteres/seg. requerida para contabilidad. Esta versatilidad permite un gran ahorro en su compra y en el trabajo diario de oficina. Incluye, además, otras características sobresalientes de una gran impresora como: la posibilidad de cambiar los tipos y tamaños de caracteres entre normal, condensado, pica, élite y proporcional, la cantidad de caracteres por línea entre 136, 272 y proporcional, los espacios entre líneas entre 1/8", 1/8", 14/120", n/60", n/120", la velocidad de alimentación de líneas entre 2,0"/seg. y 3,0"/seg., y puede alimentar papel con anchura entre 127 mm y 381 mm por fricción o continuo por medio de su tractor incorporado y con capacidad de imprimir un original y cuatro copias con un nivel de ruido inferior a 65 db A.



# COMMODORE 16 LA EMOCION DE EMPEZAR

Iniciarse en el mundo de los ordenadores personales con un COMMODORE 16 es sumar, a la emoción de empezar, la emoción del futuro.

Porque es un ordenador de fácil manejo y programación, pero con prestaciones que sólo se encuentran en ordenadores de costo mucho más elevado.

Porque es un ordenador pequeño, pero con la mayor cantidad de

software y periféricos, que multiplican sus posibilidades futuras.

Porque es el ordenador ideal para empezar y perfecto para seguir.

## PRINCIPALES CARACTERISTICAS

- |   |   |
|---|---|
| <ul style="list-style-type: none"> <li>- 16 K.</li> <li>- COMANDOS DE ALTO NIVEL PARA GESTION DE COLOR, SONIDO Y GRAFICOS DE ALTA RESOLUCION.</li> <li>- TECLADO PROFESIONAL</li> </ul> | <ul style="list-style-type: none"> <li>- 40 COLUMNAS x 25 LINEAS.</li> <li>- 121 COLORES.</li> <li>- GRAFICOS EN ALTA RESOLUCION.</li> <li>- 2 GENERADORES DE TONO.</li> <li>- AMPLIA GAMA DE PERIFERICOS.</li> </ul> |
|---|---|

COMMODORE COMPUTER. Advierte que al comprar uno de sus productos cede la garantía de MICROELECTRONICA Y CONTROL, S.A., única válida para todo el territorio nacional.

**commodore 16**  
LA EMOCION DEL FUTURO

**commodore**  
**COMPUTER**

MICROELECTRONICA Y CONTROL

c/ Valencia, 49-53. 08015 Barcelona. c/ Princesa, 47, 3.º G. 28008 Madrid.

UNICO REPRESENTANTE DE COMMODORE EN ESPAÑA